VEHICLE TRAVEL TIME DISTRIBUTION ESTIMATION AND MAP-MATCHING VIA MARKOV CHAIN MONTE CARLO METHODS

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Bradford S. Westgate

August 2013

VEHICLE TRAVEL TIME DISTRIBUTION ESTIMATION AND
MAP-MATCHING VIA MARKOV CHAIN MONTE CARLO METHODS

Bradford S. Westgate, Ph.D.
Cornell University 2013

We introduce two statistical methods for estimating vehicle travel time distributions on a road network, using Global Positioning System (GPS) data recorded during historical vehicle trips. In the first method, we use a model of the path taken by each vehicle in the data, the travel time on each road segment in the network, and the location and speed errors for each GPS observation. In the second method, we use a model of the entire travel time of each trip, and include covariates such as the types of roads traveled and time of day. We estimate the parameters of both models by Markov chain Monte Carlo methods.

We compare the performance of these methods with two simpler methods, a recently published method, and commercially available travel time estimates, using data from ambulance trips in Toronto and simulated data. Our methods outperform the alternative methods in point and distribution estimation of out-of-sample trip travel times. Our methods also provide more realistic estimates than the recently published method of the probability that an ambulance is able to respond to each intersection in Toronto within a time threshold.

We also consider map-matching, i.e. estimating a vehicle's path from sparse and error-prone GPS data, which is an important sub-problem for travel time estimation. In practice, successive GPS location readings are frequently biased in the same direction. We introduce a statistical map-matching method that takes into account bias in GPS locations, leading to improved accuracy.

BIOGRAPHICAL SKETCH

Brad grew up in Nashua, New Hampshire. He was homeschooled through high school, and spent most of his childhood playing sports and reading books. He enjoyed mathematics from a young age, and coached a homeschool math team during high school. Later he worked as a teaching assistant in the Center for Talented Youth program. These positive experiences with teaching led him to pursue becoming a professor. He attended Olin College of Engineering, graduating in 2008 with a degree in Engineering, concentrating in Computing. In 2013 he will teach probability and statistics as a visiting professor at Mount Holyoke College. He still enjoys playing sports, particularly basketball, at least at the beginning of the game before he remembers that he is out of shape. He also enjoys reading aloud and volunteering with children.

This thesis is dedicated to my grandmother Lena.

ACKNOWLEDGEMENTS

I am grateful to my committee members Dawn Woodard, David Matteson, Shane Henderson, and David Williamson for all their help and advice. I thank Dawn, Shane, and David M. for their longstanding collaboration and ideas, and I particularly thank my main advisor Dawn for her constant encouragement and detailed feedback. I thank Dave Lyons, Toronto EMS, TomTom, and The Optima Corporation for their collaboration and the use of their data, and I thank Christopher Glessner for his help with the TomTom experiments. I thank the National Science Foundation for their support of this research.

I thank my fellow Operations Research graduate students for their friendship throughout our years in Ithaca. I thank my friends in the Cornell Graduate Christian Fellowship for all our conversations and fun times. I thank my roommate Jon Steffens for being so easy to live with, and for his patience with my irregular cleaning habits. Finally, I especially thank my parents Brad and Betsy, my sister Barbara, and my friend Justin Wong for all their love and support.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Motivation

Travel time estimates for vehicles on a road network are used in navigation systems, transport policy decisions, and management of vehicle fleets such as taxis, emergency vehicles, and delivery services [10]. We are motivated particularly by the emergency medical services (EMS) application. In this context, travel times are used in algorithms for positioning ambulance bases and parking locations [7, 20, 23], in ambulance redeployment methods [38], and in ambulance dispatch decisions [11]. For example, EMS providers prefer to assign the ambulance expected to arrive fastest to respond to a new emergency [11], which requires a travel time estimate for each available ambulance to the emergency location.

In the EMS context and others, it is also important to capture the uncertainty in the travel time, by estimating the entire travel time distribution, rather than just the mean travel time [27, 48]. For instance, taking into account uncertainty in ambulance travel times can improve fleet management decisions and thereby reduce response times, leading to higher quality care for patients [12, 40]. Ambulance travel time performance targets are also framed in terms of the distribution. EMS contracts typically stipulate that the EMS organization must respond to a certain fraction of emergencies within a time threshold, or fines are assessed [13, 36]. Similarly, Pell et al. estimated that improving response times from 90% of emergencies within 14 minutes to 90% of emergencies within 8 minutes would increase the survival rate of out-of-hospital heart attack

patients from 6% to 8%, on data from Scotland [43].

In Chapters 2 and 3, we introduce and compare two statistical methods for vehicle travel time distribution estimation, using Global Positioning System data (GPS) recorded during historical vehicle trips. Our travel time estimation methods are designed particularly for ambulance data, but are applicable in other contexts. Indeed, GPS data from smartphones and other navigation devices are increasingly available from many sources, including taxi fleets, delivery services, and personal vehicles [5]. Unlike other sources of travel time data, GPS devices do not require instrumentation on the roadway, and therefore have the prospect of comprehensive network coverage [25].

Raw GPS data are subject to error in location and speed measurements [64, 65]. Location accuracy is particularly poor in urban canyons, where GPS satellites may be obscured and signals reflected [9, 36]. Large errors of over one-hundred meters are not uncommon [5, 9]. Often, GPS data are also sparsely recorded. Sparsity is introduced to reduce data transmission and storage costs [41, 46], or to save smart-phone battery life [26]. In some cases, GPS observations can be as infrequent as every 1-2 kilometers or more [34].

Sparsity and error in GPS readings can make it difficult to reconstruct the path traversed by a vehicle. Estimating a vehicle path from a set of GPS readings is called the map-matching problem [63]. Map-matching is a popular topic of current research, because of the explosion in quantity of sparse, error-prone GPS data [5, 26]. Map-matching is an important sub-problem for travel time estimation, because typically we must know the route traveled in each historical vehicle trip in order to predict travel times. In our first travel time estimation method, map-matching solutions for each vehicle trip are estimated simultane-

ously. In our second estimation method, map-matching solutions are required as inputs. In Chapter 4, we introduce a statistical map-matching method.

To test our two travel time estimation methods and compare to alternative methods, we use data provided by Toronto EMS, from the years 2007-2008. These data consist of GPS observations on ambulance trips, and exhibit sparsity and error. GPS readings are typically drawn every 200 meters of travel, though sometimes the interval is larger or smaller. The Toronto dataset contains 157,283 ambulance trips, and the road network contains 68,272 links (road segments between neighboring intersections). The large size of this dataset makes computational efficiency an important consideration for our methods.

1.2 Alternative Travel Time Estimation Methods

First, we review alternative approaches for vehicle travel time estimation. Hofleitner, Herring, and Bayen [25] and Hofleitner et al. [24] take a traffic flow perspective, modeling travel times at the network link level. They use a dynamic Bayesian network for the unobserved traffic conditions on links and model the link travel time distributions conditional on the traffic state. Their method is applied to a subset of the San Francisco road network with roughly 800 links, predicting travel times using taxi fleet data and validating with additional data sources.

Jenelius and Koutsopoulos [28] propose a framework for estimating vehicle travel time distributions while incorporating weather, speed limit, and other explanatory factors. They point out that empirical evidence suggests that the link travel times are strongly correlated, even after conditioning on time of day and

other explanatory factors [3, 48]. This contrasts with approaches such as Hofleitner et al. [24, 25] and our first method [62], which assume that the link travel times are independent within a vehicle trip, perhaps conditional on the traffic state. Jenelius and Koutsopoulos capture correlation using a moving average specification for the link travel times. Their framework is applied to estimate travel times for a particular route in Stockholm.

The conditions of these articles differ from our application. They have a higher density of data for particular times and routes in the network than exists in our Toronto ambulance data, because ambulance trips are rare compared to other vehicles. The high density of data allows Hofleitner et al. to model traffic dynamics directly [24, 25]. Although our data is less dense in time, the size of the Toronto road network that we use is an order of magnitude larger than in these articles, and the number of historical vehicle trips is also larger [24, 25, 28, 61]. This leads to different modeling choices and computational challenges.

Research on estimating specifically ambulance travel time distributions has been done by Budge, Ingolfsson and Zerom [8]. They model ambulance travel times using a log t-distribution, where the median and coefficient of variation are either nonparametric or parametric functions of the shortest-path distance between the start and end locations [32]. These functional forms enable their method to be flexible but still interpretable. However, the reliance on trip distance means that their method cannot capture some desired features, such as faster response times to locations near major roads. We compare travel time estimation performance with the method of Budge et al. in Chapters 2 and 3.

Aladdini [1] investigated ambulance travel time distributions between specific start and end locations in Waterloo, Ontario. He found that the travel times

were well modeled by lognormal distributions, in contrast to Budge et al., who observed heavier tails [8]. We also find that the lognormal distribution provides a good fit (Section 3.3.2). Part of this difference appears to be because Budge et al. do not condition on the trip location; all trips of the same length are treated together. We desire to go beyond Aladdini and model travel time distributions for arbitrary routes. Ambulances are often assigned to new emergencies while away from their bases [11], and so we need richer information than estimated response time distributions from several fixed bases.

In addition to these studies, there are also commercially-available travel time estimates. Specifically, we investigate travel time estimates from TomTom, a maker of navigation products (Section 3.4). Their travel time estimates are based on data from TomTom navigation devices, and provide only mean travel times, and so cannot be used in applications where travel time distributions are required. Also, their estimates are calculated for standard vehicle speeds, not "lights-and-sirens" ambulance speeds. However, they are still useful for point estimation performance comparisons, as long as they are corrected for bias.

TomTom generates travel time estimates using both real-time and historical GPS information. Other real-time sources of travel information are available, for example from Google and Waze. In this thesis, we rely on historical ambulance data. However, as these real-time data sources become more comprehensive and EMS organizations make use of them, it will likely become beneficial to integrate real-time and historical data for ambulance travel time estimation. We discuss this as an area for further study in Chapter 5.

1.3 Independent Link Estimation Method and Local Methods

We introduce our first vehicle travel time distribution estimation method in Chapter 2. We use a statistical model on the distribution of GPS location and speed errors, the path traveled by each vehicle, and the travel time on each network link (Section 2.2). The model combines information from the GPS times, locations, and speeds observed during each historical vehicle trip with the start and end times and locations of the trips. To simplify analysis, we assume independence between the travel times on each link and between all the GPS speed and location errors. We refer to this method as the Independent Link (IL) method.

To estimate the parameters of the model, we take a Bayesian perspective and introduce a Markov chain Monte Carlo method to draw samples from the posterior distribution of the unknown parameters [55, 56] (Section 2.3). This simultaneous estimation allows uncertainty in each parameter (for example, the path traveled in each trip) to be taken into account in estimating the other parameters. To sample the path traveled by each vehicle, we introduce a reversible jump Metropolis-Hastings proposal [21]. The reversible jump proposal is given in Section 2.3.2 and its validity proven in Section 2.3.7.

We also introduce two local methods using only the GPS locations and speeds (Section 2.4.1). Each GPS reading is assigned to the nearest link, and the GPS speeds are used to estimate the travel time distribution for each link. The local methods are straightforward, requiring no map-matching solutions or sophisticated modeling, and provide helpful comparisons to our other methods. Also, they are useful in settings where more sophisticated models require

initial speed or time estimates for the roads in a network [34, 60, 61].

In the first local method, we use the harmonic mean of the mapped GPS speeds to create a point estimate of the travel time. We are the first to propose this estimator for GPS data, though it is commonly used in the transportation literature for estimating travel times via speed data recorded by loop detectors [47, 53, 58]. We show that if the GPS readings are sampled by distance (i.e. every 200 meters), then this method is unbiased; however if the GPS readings are sampled by time (i.e. every 10 seconds), then this method overestimates the mean travel time (Section 2.4.2). This method also naturally produces a travel time distribution estimate. In the second local method, we assume a parametric distribution for the GPS speeds on each link, and calculate maximum likelihood estimates of the distribution parameters, which can be used to obtain point and distribution estimates of the travel time.

We compare the out-of-sample trip travel time predictive accuracy of the IL method, the local methods, and the method of Budge et al. on the subregion of Leaside, Toronto. Point estimates from the IL method outperform the alternative methods by 1-5% in root mean squared error (RMSE) on the Toronto ambulance data and by 4-8% in RMSE on simulated data (Sections 2.6 and 2.7). We also introduce an Oracle method to calculate the amount of unavoidable prediction error due to random travel times, even when the true distribution is known exactly. If the unavoidable error is subtracted, the IL method outperforms the alternative methods by over 50% in RMSE on the simulated data.

For travel time distribution estimation, we calculate 95% predictive intervals from each method. Intervals from the IL method on the Toronto ambulance data are narrower than those from Budge et al., which is desirable. However, only

85.8% of the observed travel times are contained in the 95% predictive intervals from the IL method, indicating that the intervals do not capture the full range of travel time variability. This is probably because the assumption of independence between link travel times does not hold in practice [3, 48]. Dependence between link travel times leads to greater variability in trip travel times.

We also calculate the probability that an ambulance is able to travel from a start location to each intersection in the Toronto subregion within a specific time threshold (Section 2.7.4). These probabilities are applied in the EMS travel time performance targets mentioned above. Visual displays of these probabilities are called probability-of-coverage maps, and are useful to EMS practitioners [8]. The estimated probabilities from the IL method are higher for locations that can be reached by fast roads than for locations that are the same distance from the start location but cannot be reached by fast roads. This behavior cannot be captured by the method of Budge et al., and so the estimated probabilities from our method appear more realistic.

Finally, we assess the ambulance path estimates from the IL method as solutions to the map-matching problem (Sections 2.6.3 and 2.7.5). The posterior distribution from the IL method is able to capture multiple high-probability paths when the true path is unclear from the GPS data. Path estimates from the IL method interpolate accurately between widely-separated GPS locations and are robust to GPS error.

1.4 Whole Trip Estimation Method

Although our IL method is successful in estimating ambulance travel times on the Toronto subregion, there is room for improvement in estimation performance and computational requirements. The method is computationally intensive, primarily because there are a large number of parameters to be estimated. Also, the assumption of independence between link travel times leads to travel time interval estimates that are unrealistically narrow, as discussed above.

We address these difficulties by proposing a statistical model on the whole travel time for each trip, rather than on the individual link travel times. This naturally incorporates dependence between link travel times. We refer to this method as the Whole Trip (WT) travel time estimation method. Like Budge et al. [8], we estimate the trip travel time via a parametric model, but our model also incorporates dependence on the route taken and other explanatory factors. The WT method uses a flexible model of the parameters of the trip travel time distribution, given total travel times and estimated paths from historical trips on the network. In order to predict the travel time distribution for a particular path, the model does not require historical trips that take precisely the same path. Instead, it uses information from all the historical trips by learning shared properties like the effects of time of day and types of road traversed.

Specifically, the WT model uses parameters for the unit travel time (inverse of speed) for each road class (highway, major arterial, etc.) in the network, parameters for each time bin of the week, and parameters relating the travel time variability to the distance traveled. These modeling choices are suggested by exploratory data analysis (Section 3.3.2). For computation, we again take a

Bayesian perspective and introduce a Markov chain Monte Carlo method to estimate the model parameters (Section 3.2.2).

The WT method is more computationally efficient than the IL method. The number of parameters in the IL method grows with the number of links in the network, the number of paths in the dataset, and the number of links taken in each path. The number of parameters in the WT method is invariant to these quantities. Each parameter must be estimated, and a large number of parameters may also increase the number of iterations required to converge to the limiting distribution of the Markov chain.

However, the WT method does make some modeling simplifications. It does not estimate map-matching solutions for the historical vehicle trips, but requires them as inputs. Thus, uncertainty in the path traveled by each vehicle is lost in the travel time estimation stage. If there is a large amount of uncertainty in some of the paths, given the GPS data, this could have a negative effect on estimation performance. In the Toronto ambulance data, the path is clear for many of the trips, but there are also many trips where at least part of the path is unclear. Also, the WT method only uses the times of the first and last GPS readings (after the map-matching inputs are generated); the interior GPS readings are ignored. This leads to loss of information about individual link travel times and how the vehicle travels during each trip, compared to the IL method. It is possible to form a model that includes all the GPS data and also retains some of the advantages of the WT method, such as dependence between link travel times. We discuss this as an area for future work in Chapter 5.

We use the WT method to predict travel times for out-of-sample ambulance trips for the entire Toronto dataset, and compare the prediction accuracy to that

of Budge et al. [8] and the TomTom estimates (Section 3.5). We consider two scenarios: (1) where the path traveled for each test trip is assumed known, and (2) where the path is assumed unknown and estimated via the fastest path in expected travel time. Point estimates from the WT method outperform Budge et al. by 3.5% in RMSE under Scenario 1 and by 2.5% under Scenario 2, and outperform TomTom by 5% under Scenario 2, which is the fairer comparison because we do not specify the paths traveled when obtaining the TomTom estimates. Performance of both the WT method and Budge et al. improves substantially from Scenario 2 to Scenario 1, indicating that travel time predictions can be more accurate if the path traveled is specified in advance. For distribution estimation, the WT method outperforms Budge et al. by 3% in continuous ranked probability score [18]. We also compare performance with the IL method on the subregion of Toronto used in Chapter 2. The WT method performs comparably to the IL method in point estimation and better in interval estimation.

We also compare the WT method with the method of Budge et al. in terms of their effect on ambulance fleet management. We select a set of representative ambulance posts in Toronto, and calculate which ambulance post is estimated to be the closest in median travel time to each intersection in Toronto, according to the two methods. We find that 5% of the intersections in the city have different estimated closest posts according to the two methods, and therefore the methods would recommend that a different ambulance respond to emergencies at these intersections, if the closest ambulance is dispatched [11]. We also calculate the probability that an ambulance is able to respond within 4 minutes from the closest post to each intersection in the city. We find substantial disparities between the two methods; for 10% of the intersections in the city, the two methods give response probabilities that differ by at least 15%. As in Chapter 2,

these disparities appear to arise because our method allows differences in speed between roads, unlike the method of Budge et al.

1.5 Map-Matching and GPS Location Bias Estimation

In Chapter 4, we introduce a statistical map-matching method. First, we review recent approaches to map-matching [26, 34, 41, 46]. Most map-matching algorithms return a single best estimate of the path driven by the vehicle [5]. However, some applications such as route choice models use a set of possible paths with associated probabilities [5]. Bierlaire, Chen and Newman [5] introduced a map-matching method that returns a probability for each candidate path. Our IL method performs map-matching and gives a posterior probability for each path, as does our map-matching method introduced below.

It has been observed that successive GPS location errors appear to be dependent, in the form of a persistent bias in a particular direction, together with a smaller independent random noise [31, 66]. Xu et al. observed that the GPS bias was fairly stable in the short term and changed smoothly on the time-scale of minutes [66]. There are several reasons why GPS locations are biased. These include apparent biases due to errors and simplifications in the digital road network [9], such as the typical assumption that roads are sequences of line segments with no width, and inherent properties of the GPS system, such as atmospheric delay [31] and the use of dead-reckoning in cases where GPS satellites cannot be observed [66]. GPS bias and random noise have been corrected for via Kalman filters in the high-frequency GPS setting [31, 66]. However, in map-matching methods for sparse GPS data, location errors are typically assumed to

be independent and normally distributed [5, 26, 33, 34, 36, 62].

In Chapter 4, we first investigate whether the path traversed and the GPS location bias are identifiable, i.e. whether they can be estimated uniquely given sufficient data (Section 4.2). In the case where there is no independent error for each reading, we show that the path and bias are identifiable up to translations of the path in the road network by a shift vector. However, even if there is no path in the road network that is a translation of the true path, the true path and GPS bias may not be distinguishable from alternatives given only a finite amount of GPS data.

Next, we introduce a statistical map-matching method that models the GPS location error as the sum of a bias vector for the entire trip and an independent error for each reading (Section 4.3). We simultaneously estimate map-matching solutions and the GPS bias and independent error distributions for a dataset of historical vehicle trips, using Bayesian methods. We compare the Metropolis-Hastings proposal we use to sample paths on a road network with a similar method for estimating paths recently introduced by Flötteröd and Bierlaire [14].

We test our map-matching method on the Toronto ambulance data and on simulated data, comparing the method to a reduced method that does not include a term for the GPS bias (Sections 4.4 and 4.5). We find that the method that includes bias outperforms the reduced method on simulated datasets where the GPS bias is medium-to-large and the independent error is small. The two methods perform comparably when both types of errors are small, and the reduced method performs slightly better when the independent errors are large. In real data, it appears that the independent error is almost always small, and the biases range from small to large. We also investigate specific types of paths in the

Toronto ambulance data and the simulated data in which the model including bias performs better.

1.6 Summary of Remaining Chapters

In Chapter 2, we introduce our IL travel time estimation method and local methods, and make comparisons to the method of Budge et al. [8] on a subregion of Toronto. Chapter 2 and material in this Introduction were published in The Annals of Applied Statistics [62]. In Chapter 3, we introduce our WT estimation method, and make comparisons to the IL method, the method of Budge et al., and the TomTom estimates on the entire Toronto dataset. This chapter and material in this Introduction have been submitted for publication [61]. In Chapter 4, we introduce our map-matching and bias estimation method. This chapter is a working paper and appears here for the first time [60]. We draw conclusions and consider areas for future work in Chapter 5.

CHAPTER 2

TRAVEL TIME ESTIMATION FOR AMBULANCES USING BAYESIAN DATA AUGMENTATION

2.1 Introduction

Emergency medical service (EMS) providers prefer to assign the closest available ambulance to respond to a new emergency [11]. Thus, it is vital to have accurate estimates of the travel time of each ambulance to the emergency location. An ambulance is often assigned to a new emergency while away from its base [11], so the problem is more difficult than estimating response times from several fixed bases. Travel times also play a central role in positioning bases and parking locations [7, 20, 23]. Accounting for variability in travel times can lead to considerable improvements in EMS management [12, 27]. We introduce methods for estimating the distribution of travel times for arbitrary routes on a municipal road network, using historical trip durations and vehicle Global Positioning System (GPS) readings. This enables estimation of fastest paths in expectation between any two locations, as well as estimation of the probability an ambulance will reach its destination within a given time threshold.

Most EMS providers record ambulance GPS information; we use data from Toronto EMS from 2007-2008. The GPS data include locations, timestamps, speeds, and vehicle and emergency incident identifiers. Readings are stored every 200 meters (m) or 240 seconds (s), whichever comes first. The true sampling rate is higher, but this scheme minimizes data transmission and storage. This is standard practice across EMS providers, though the storage rates vary [36]. In related applications the GPS readings can be even sparser; Lou et al.

[34] analyzed data from taxis in Tokyo in which GPS readings are separated by 1-2 km or more.

The GPS location and speed data are also subject to error. Location accuracy degrades in urban canyons, where GPS satellites may be obscured and signals reflected [9, 36]. Chen et al. [9] observed average location errors of 27 m in parts of Hong Kong with narrow streets and tall buildings, with some errors over 100 m. Location error is also present in the Toronto data; see Figure 2.1. Witte and Wilson [65] found GPS speed errors of roughly 5% on average, with largest error at high speeds and when few GPS satellites were visible.

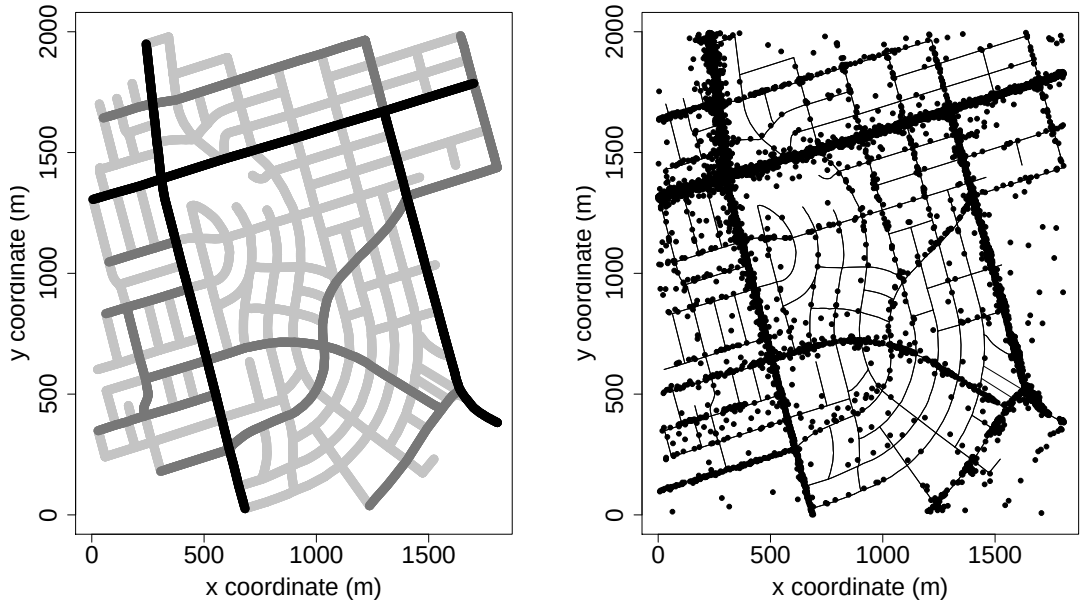

Figure 2.1: Left: A subregion of Toronto, with primary roads (black), secondary roads (gray) and tertiary roads (light gray). Right: GPS data on this region from the Toronto EMS lights-and-sirens dataset.

Recent work on estimating ambulance travel time distributions has been done by Budge, Ingolfsson and Zerom [8] and Aladdini [1], using estimates based on total trip distance and time, not GPS data. Budge et al. proposed mod-

eling the log travel times using a t-distribution, where the median and coefficient of variation are functions of the trip distance (see Section 2.4.3). Aladdini found that the lognormal distribution provided a good fit for ambulance travel times between specific start and end locations. Budge et al. found heavier tails than Aladdini, in part because they did not condition on the trip location.

We first introduce two local methods using only the GPS locations and speeds (Section 2.4.1). Each GPS reading is assigned to the nearest link (the section of road between neighboring intersections), and the assigned speeds are used to estimate the travel time for each link. In the first method, we use the harmonic mean of the mapped GPS speeds to create a point estimator of the travel time. We are the first to propose this estimator for mapped GPS data, though it is commonly used for estimating travel times via speed data recorded by loop detectors [47, 53, 58]. We give theoretical results supporting this approach in Section 2.4.2. This method also yields interval and distribution estimates of the travel time. In our second local method, we assume a parametric distribution for the GPS speeds on each segment, and calculate maximum likelihood estimates of the parameters of this distribution. These can be used to obtain point, interval, or distribution estimates of the travel time.

In Sections 2.2 and 2.3, we propose a more sophisticated method, modeling the data at the trip level. Whereas the local methods use only GPS data and the method of Budge et al. uses only the trip start and end locations and times, this method combines the two sources of information. We simultaneously estimate the path driven for each ambulance trip and the distribution of travel times on each link, using Bayesian data augmentation [55]. For computation, we introduce a reversible jump Markov chain Monte Carlo method [21]. We refer to this

method as the Independent Link (IL) method, since the model assumes independence between link travel times.

We compare the predictive accuracy on out-of-sample trips for the IL method, the local methods, and the method of Budge et al. on a subregion of Toronto, using simulated data and real data (Sections 2.6 and 2.7). Since the methods have some bias due in part to the GPS sampling scheme, we use a correction factor to make each method approximately unbiased (Section 2.5). On simulated data, point estimates from the IL method outperform the alternative methods by over 50% in root mean squared error, relative to an Oracle method with the lowest possible error. On real data, point estimates from the IL method again outperform the alternative methods. Interval estimates from the IL method are superior to those from the local methods, but appear to be slightly too narrow to capture the full range of travel time variability.

We also produce probability-of-coverage maps [8], showing the probability of traveling from a given intersection to any other intersection within a time threshold (Section 2.7.4). This is the performance standard in many EMS contracts; an EMS organization attempts to respond to, e.g., 90% of all emergencies within 9 minutes [13]. The estimates from the IL method are more realistic than those of Budge et al., because they differentiate between equidistant locations based on whether or not they can be reached by fast roads.

Finally, we assess the ambulance path estimates from the IL method (Sections 2.6.3 and 2.7.5). Estimating the path driven from a discrete set of GPS readings is called the map-matching problem [36]. Most map-matching algorithms return a single path estimate [33, 34, 35, 36]. However, the posterior distribution of the IL method can capture multiple high-probability paths when

the true path is unclear from the GPS data. Path estimates from the IL method interpolate accurately between widely-separated GPS locations and are robust to GPS error.

2.2 Bayesian Formulation

2.2.1 Model

Consider a network of J directed road segments, called links, and a set of I ambulance trips on this network. Assume that each trip starts and ends on known nodes (intersections) d_i^s and d_i^f in the network, at known times t_i^s and t_i^f. Therefore the total travel time $t_i^f - t_i^s$ is known. In practice, trips sometimes begin or end in the interior of a link; however, links are short enough that this is a minor issue; the median link length in the full Toronto network is 111 m, the mean is 162 m, and the maximum is 4613 m. Each trip i has observed GPS readings, indexed by $\ell \in \{1, \ldots, r_i\}$, and gathered at known times t_i^ℓ. GPS reading ℓ is the triplet $\left(X_i^\ell, Y_i^\ell, V_i^\ell\right)$, where X_i^ℓ and Y_i^ℓ are the measured geographic coordinates and V_i^ℓ is the measured speed. Denote $G_i = \left\{\left(X_i^\ell, Y_i^\ell, V_i^\ell\right)\right\}_{\ell=1}^{r_i}$.

The relevant unobserved variables for each trip i are the following:

1. The unknown path (sequence of links) $A_i = \{A_{i,1}, \ldots, A_{i,N_i}\}$ traveled by the ambulance from d_i^s to d_i^f. The path length N_i is also unknown.
2. The unknown travel times $T_i = (T_{i,1}, \ldots, T_{i,N_i})$ on the links in the path. We use the notation $T_i(j)$ to refer to the travel time in trip i on link j.

We model the observed and unobserved variables $\{A_i, T_i, G_i\}_{i=1}^I$ as follows. Conditional on A_i, each element $T_{i,k}$ of the vector T_i follows a lognormal distribution with parameters $\mu_{A_{i,k}}, \sigma^2_{A_{i,k}}$, independently across i and k. We use the notation $T_{i,k}|A_i \sim \mathcal{LN}\left(\mu_{A_{i,k}}, \sigma^2_{A_{i,k}}\right)$. In the literature, ambulance travel times between specific locations have been observed and modeled to be lognormal [1, 2]. Denote the expected travel time on each link $j \in \{1, \ldots, J\}$ by $\theta(j) = \exp\left(\mu_j + \sigma_j^2/2\right)$. We use a multinomial logit choice model [39] for the path A_i, with likelihood

$$f(A_i) = \frac{\exp\left(-C\sum_{k=1}^{N_i}\theta\left(A_{i,k}\right)\right)}{\sum_{a_i \in \mathcal{P}_i}\exp\left(-C\sum_{k=1}^{n_i}\theta\left(a_{i,k}\right)\right)}, \tag{2.1}$$

where $C > 0$ is a fixed constant, $\mathcal{P}_i$ is the set of possible paths with no repeated nodes from d_i^s to d_i^f in the network, and $a_i = \{a_{i,1}, \ldots, a_{i,n_i}\}$ indexes the paths in $\mathcal{P}_i$. In this model, the fastest routes in expectation have the highest probability, and the ratio of probabilities between two routes is a function of their difference in expected travel time.

We assume that ambulances travel at constant speed on a single link in a given trip. This approximation is necessary since there is typically at most one GPS reading on any link in a given trip, and thus little information in the data regarding changes in speed on individual links. Therefore, the true location and speed of the ambulance at time t_i^ℓ are deterministic functions $\text{loc}\left(A_i, T_i, t_i^\ell\right)$ and $\text{sp}\left(A_i, T_i, t_i^\ell\right)$ of A_i and T_i. Conditional on A_i, T_i, the measured location $\left(X_i^\ell, Y_i^\ell\right)$ is assumed to have a bivariate normal distribution (a standard assumption [33, 36]) centered at $\text{loc}\left(A_i, T_i, t_i^\ell\right)$, with known covariance matrix Σ. Similarly, the measured speed V_i^ℓ is assumed to have a lognormal distribution with

expectation equal to sp $\left(A_i, T_i, t_i^\ell\right)$ and variance parameter ζ^2:

$$\left(X_i^\ell, Y_i^\ell\right) \Big| A_i, T_i \sim N_2\left(\text{loc}\left(A_i, T_i, t_i^\ell\right), \Sigma\right), \tag{2.2}$$

$$\log V_i^\ell \Big| A_i, T_i \sim N\left(\log \text{sp}\left(A_i, T_i, t_i^\ell\right) - \frac{\zeta^2}{2}, \zeta^2\right). \tag{2.3}$$

We assume independence between all the GPS speed and location errors. Combining Equations 2.1-2.3, we obtain the complete-data likelihood

$$f\left(\left\{A_i, T_i, G_i\right\}_{i=1}^I \Big| \left\{\mu_j, \sigma_j^2\right\}_{j=1}^J, \zeta^2\right) = \prod_{i=1}^I \left[f(A_i) \prod_{k=1}^{N_i} \mathcal{LN}\left(T_{i,k}; \mu_{A_{i,k}}, \sigma_{A_{i,k}}^2\right) \right. \tag{2.4}$$
$$\left. \prod_{\ell=1}^{r_i} \left[N_2\left(\left(X_i^\ell, Y_i^\ell\right); \text{loc}\left(A_i, T_i, t_i^\ell\right), \Sigma\right) \times \mathcal{LN}\left(V_i^\ell; \log \text{sp}\left(A_i, T_i, t_i^\ell\right) - \frac{\zeta^2}{2}, \zeta^2\right) \right] \right].$$

In practice we use data-based choices for the constants Σ and C (see Section 2.3.6). The unknown parameters in the model are the link travel time parameters $\left\{\mu_j, \sigma_j^2\right\}_{j=1}^J$ and the GPS speed error parameter ζ^2.

2.2.2 Prior Distributions

We specify independent prior distributions for the unknown parameters, using $\mu_j \sim N\left(m_j, s^2\right)$, $\sigma_j \sim \text{Unif}\left(b_1, b_2\right)$, and $\zeta \sim \text{Unif}\left(b_3, b_4\right)$, where m_j, s^2, b_1, b_2, b_3, b_4 are fixed hyperparameters. A normal prior is a standard choice for the location parameter of a lognormal distribution. We use uniform priors on the standard deviations σ_j and ζ [15]. The prior ranges $[b_1, b_2]$ and $[b_3, b_4]$ are made wide enough to capture all plausible parameter values. The prior mean for μ_j depends on j, because there are often existing road speed estimates that can be used to specify m_j. Prior information regarding the values s^2, b_1, b_2, b_3, b_4 is more limited. We use a combination of prior information and the data to specify all hyperparameters, as described in Section 2.3.6.

2.3 Bayesian Computational Method

We use a Markov chain Monte Carlo method to obtain samples from the joint posterior distribution of all unknowns [51, 56]. Each unknown is updated in turn, conditional on the other unknowns, via either a draw from the closed-form conditional posterior distribution or a Metropolis-Hastings (M-H) move. Estimation of any desired function $g\left(\zeta^2, \left\{\mu_j, \sigma_j^2\right\}_{j=1}^J\right)$ of the unknown parameters is done via the Monte Carlo samples $\left(\zeta^{2(\ell)}, \left\{\mu_j^{(\ell)}, \sigma_j^{(\ell)}\right\}_{j=1}^J, \left\{A_i^{(\ell)}, T_i^{(\ell)}\right\}_{i=1}^I\right)$, taking $\hat{g} = \frac{1}{M}\sum_{\ell=1}^M g\left(\zeta^{2(\ell)}, \left\{\mu_j^{(\ell)}, \sigma_j^{2(\ell)}\right\}_{j=1}^J\right)$.

2.3.1 Markov Chain Initial Conditions

To initialize each path A_i, select the middle GPS reading, reading number $\lfloor r_i/2 \rfloor + 1$. Find the nearest node in the road network to this GPS location, and route the initial path A_i through this node, taking the shortest-distance path to and from the middle node. To initialize the travel time vector T_i, distribute the known trip time across the links in the path A_i, weighted by link length. Finally, to initialize ζ^2 and each μ_j and σ_j^2, draw from their priors.

2.3.2 Updating the Paths

Updating the path A_i may also require updating the travel times T_i, since the number of links in the path may change. Since this changes the dimension of the vector T_i, we update (A_i, T_i) using a reversible jump M-H move [21]. Calling the current values $\left(A_i^{(1)}, T_i^{(1)}\right)$, we propose new values $\left(A_i^{(2)}, T_i^{(2)}\right)$ and accept

them with the appropriate probability, detailed below.

The proposal changes a contiguous subset of the path. The length (number of links) of this subpath is limited to some maximum value K; we specify K in Section 2.3.5. Precisely:

1. With equal probability, choose a node d' from the path $A_i^{(1)}$, excluding the final node.
2. Let $a^{(1)}$ be the number of nodes that follow d' in the path. With equal probability, choose an integer $w \in \{1, \ldots, \min\left(a^{(1)}, K\right)\}$. Denote the wth node following d' as d''. The subpath from d' to d'' is the section to be updated (the "current update section").
3. Consider all possible routes of length up to K from d' to d''. With equal probability, propose one of these routes as a change to the path (the "proposed update section"), giving the proposed path $A_i^{(2)}$.

Next we propose travel times $T_i^{(2)}$ that are compatible with $A_i^{(2)}$. Let $\{c_1, \ldots, c_m\} \subset A_i^{(1)}$ and $\{p_1, \ldots, p_n\} \subset A_i^{(2)}$ denote the links in the current and proposed update sections, noting that m and n may be different. Recall that $T_i(j)$ denotes the travel time of trip i on link j. For each link $j \in A_i^{(2)} \setminus \{p_1, \ldots, p_n\}$, set $T_i^{(2)}(j) = T_i^{(1)}(j)$. Let $S_i = \sum_{\ell=1}^{m} T_i^{(1)}(c_\ell)$ be the total travel time of the current update section. Since the total travel time of the entire trip is known (see Section 2.2.1), S_i is fixed and known as well, conditional on the travel times for the links that are unchanged by this update. Therefore we must have $\sum_{\ell=1}^{n} T_i^{(2)}(p_\ell) = S_i$. The travel times $T_i^{(2)}(p_1), \ldots, T_i^{(2)}(p_n)$ are proposed by drawing $(r_1, \ldots, r_n) \sim \text{Dirichlet}\left(\alpha\theta(p_1), \ldots, \alpha\theta(p_n)\right)$ for a constant $\alpha > 0$ (specified below), and setting $T_i^{(2)}(p_\ell) = r_\ell S_i$ for $\ell \in \{1, \ldots, n\}$. The expected value of

the proposed travel time on link p_ℓ is $E\left(T_i^{(2)}(p_\ell)\right) = S_i \frac{\theta(p_\ell)}{\sum_{k=1}^{n}\theta(p_k)}$. Therefore, the expected values of the proposed times are weighted by the link travel time expected values [16]. The constant α controls the variances and covariances of the components $T_i^{(2)}(p_\ell)$. In our experience $\alpha = 1$ works well; the constant α can also be tuned to obtain a desired acceptance rate for a particular dataset [51, 52].

Let $N_i^{(j)}$ be the number of edges in the path $A_i^{(j)}$ for $j \in \{1, 2\}$, and let $a^{(2)}$ be the number of nodes that follow d' in the path $A_i^{(2)}$. We accept the proposal $(A_i^{(2)}, T_i^{(2)})$ with probability equal to the minimum of one and

$$\frac{f_i\left(A_i^{(2)}, T_i^{(2)}, G_i \left| \left\{\mu_j, \sigma_j^2\right\}_{j=1}^{J}, \zeta^2\right.\right)}{f_i\left(A_i^{(1)}, T_i^{(1)}, G_i \left| \left\{\mu_j, \sigma_j^2\right\}_{j=1}^{J}, \zeta^2\right.\right)} \times \frac{N_i^{(1)} \min(a^{(1)}, K)}{N_i^{(2)} \min(a^{(2)}, K)} \times \frac{\text{Dir}\left(\frac{T_i^{(1)}(c_1)}{S_i}, \ldots, \frac{T_i^{(1)}(c_m)}{S_i}; \alpha\theta(c_1), \ldots, \alpha\theta(c_m)\right)}{\text{Dir}\left(\frac{T_i^{(2)}(p_1)}{S_i}, \ldots, \frac{T_i^{(2)}(p_n)}{S_i}; \alpha\theta(p_1), \ldots, \alpha\theta(p_n)\right)} S_i^{n-m}, \tag{2.5}$$

where f_i is the contribution of trip i to Equation 2.4 and $\text{Dir}(x; y)$ denotes the Dirichlet density with parameter vector y, evaluated at x. The proposal density for the travel times $T_i^{(2)}(p_1), \ldots, T_i^{(2)}(p_n)$ requires a change of variables from the Dirichlet density. This leads to the factor S_i^{n-m} in the ratio of proposal densities. In Section 2.3.7, we show that this move is valid since it is reversible with respect to the conditional posterior distribution of (A_i, T_i).

2.3.3 Updating the Trip Travel Times

To update the realized travel time vector $T_i(j)$, we use the following M-H move. Given current travel times $T_i^{(1)}$, we propose travel times $T_i^{(2)}$.

1. With equal probability, choose a pair of distinct links j_1 and j_2 in the path

A_i. Let $S_i = T_i^{(1)}(j_1) + T_i^{(1)}(j_2)$.

2. Draw $(r_1, r_2) \sim \text{Dirichlet}\left(\alpha'\theta(j_1), \alpha'\theta(j_2)\right)$. Set $T_i^{(2)}(j_1) = r_1 S_i$ and $T_i^{(2)}(j_2) = r_2 S_i$.

Similarly to the path proposal above, this proposal randomly distributes the travel time over the two links, weighted by the expected travel times $\theta(j_1)$ and $\theta(j_2)$, with variances controlled by the constant α' [16]. In our experience $\alpha' = 0.5$ is effective for our application. It is straightforward to calculate the M-H acceptance probability.

2.3.4 Updating the Parameters μ_j, σ_j^2, and ζ^2

To update each μ_j, we sample from the full conditional posterior distribution, which is available in closed form. We have $\mu_j \left| \sigma_j^2, \{A_i, T_i\}_{i=1}^I \sim N\left(\hat{\mu}_j, \hat{s}_j^2\right)\right.$, where

$$\hat{s}_j^2 = \left[\frac{1}{s^2} + \frac{n_j}{\sigma_j^2}\right]^{-1}, \qquad \hat{\mu}_j = \hat{s}_j^2 \left[\frac{m_j}{s^2} + \frac{1}{\sigma_j^2}\sum_{i \in I_j} \log T_i(j)\right],$$

the set $I_j \subset \{1, \ldots, I\}$ indicates the subset of trips using link j, and $n_j = |I_j|$.

To update each σ_j^2, we use a local M-H step [56]. We propose $\sigma_j^{2*} \sim \mathcal{LN}(\log \sigma_j^2, \eta^2)$, having fixed variance η^2. The M-H acceptance probability p_σ is the minimum of 1 and

$$\frac{\sigma_j}{\sigma_j^*} \mathbf{1}_{\{\sigma_j^* \in [b_1, b_2]\}} \left(\frac{\prod_{i \in I_j} \mathcal{LN}\left(T_i(j); \mu_j, \sigma_j^{2*}\right)}{\prod_{i \in I_j} \mathcal{LN}\left(T_i(j); \mu_j, \sigma_j^2\right)}\right) \frac{\mathcal{LN}\left(\sigma_j^2; \log\left(\sigma_j^{2*}\right), \eta^2\right)}{\mathcal{LN}\left(\sigma_j^{2*}; \log\left(\sigma_j^2\right), \eta^2\right)}.$$

To update ζ^2, we use another M-H step with a lognormal proposal, with variance ν^2. The proposal variances η^2, ν^2 are tuned to achieve an acceptance rate of approximately 23% [52].

2.3.5 Markov Chain Convergence

The transition kernel for updating the path A_i is irreducible, and hence valid [56], if it is possible to move between any two paths in $\mathcal{P}_i$ in a finite number of iterations, for all i. For a given road network, the maximum update section length K can be set high enough to meet this criterion. However, the value of K should be set as low as possible, because increasing K tends to lower the acceptance rate. If there is a region of the city with sparse connectivity, the required value of K may be impractically large. For example, there could be a single link of a highway alongside many links of a parallel minor road. Then, a large K would be needed to allow transitions between the highway and the minor road. If K is kept smaller, the Markov chain is reducible. In this case, the chain converges to the posterior distribution restricted to the closed communicating class in which the chain is absorbed. If this class contains much of the posterior mass, as might arise if the initial path follows the GPS data reasonably closely, then this should be a good approximation.

In Sections 2.6 and 2.7, we apply the IL method to simulated data and data from Toronto EMS, on a subregion of Toronto with 623 links. Each Markov chain was run for 50,000 iterations (where each iteration updates all parameters), after a burn-in period of 25,000 iterations. We calculated Gelman-Rubin diagnostics [17], using two chains, for the parameters ζ^2, μ_j, and σ_j^2. Results from a typical simulation study were: potential scale reduction factor of 1.06 for ζ^2, of less than 1.1 for μ_j for 549 links (88.1%), between 1.1-1.2 for 43 links (6.9%), between 1.2-1.5 for 30 links (4.8%), and less than 2 for the remaining one link, with similar results for the parameters σ_j^2. These results indicate no lack of convergence.

Each Markov chain run for these experiments takes roughly 2 hours on a

3.2 GHz workstation. Each iteration of the Markov chain scales linearly in time with the number of links and the number of ambulance trips: $O(J + I)$, assuming the lengths of the ambulance paths do not grow as well. This assumption is reasonable, since long ambulance paths are undesirable for an EMS provider. It is much more difficult to assess how the number of iterations required for convergence changes with J and I, since this would require bounding the spectral gap of the Markov chain. The full Toronto road network has roughly 110 times as many links as the test region, and the full Toronto EMS dataset has roughly 80 times as many ambulance trips.

In practice, parameter estimates are updated infrequently and off-line. Once parameter estimation is done, prediction for new routes and generation of our figures is very fast. If parameter estimation for the IL method is computationally impractical for the entire city, it can be divided into multiple regions and estimated in parallel. We envision creating overlapping regions and discarding estimates on the boundary, to eliminate edge effects (see Section 2.7.1). During parameter estimation, trips traveling through multiple regions would be divided into portions for each region, as we have done in our Toronto EMS experiments. However, prediction for such a trip can be handled directly, given the parameter estimates for all links in the city. The fastest path in expectation may be calculated using a shortest path algorithm over the entire road network, which gives a point estimate of the trip travel time. A distribution estimate of the travel time can be obtained by sampling travel times on the links in this fastest path (see Section 2.7.3).

2.3.6 Constants and Hyperparameters

There are several constants and hyperparameters to be specified in the IL model. To set the GPS position error covariance matrix Σ, we calculate the minimum distance from each GPS location in the data to the nearest link. Assuming that the error is radially symmetric, that the vehicle was on the nearest link when it generated the GPS point, and approximating that link locally by a straight line, this minimum distance should equal the absolute value of one component of the 2-dimensional error, i.e. the absolute value of a random variable $\mathcal{E}_1 \sim N(0, \sigma^2)$, where $\Sigma = \left(\begin{smallmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{smallmatrix}\right)$. Since $E(|\mathcal{E}_1|) = \sigma\sqrt{2/\pi}$, we take $\hat{\sigma} = \hat{E}(|\mathcal{E}_1|)\sqrt{\pi/2}$, where $\hat{E}(|\mathcal{E}_1|)$ is the mean minimum distance of each GPS point to the nearest link in the data. In the Toronto EMS datasets (see Section 2.7.1), we have $\hat{E}(|\mathcal{E}_1|) = 8.4$ m for lights-and-sirens (L-S) data and 7.7 m for standard travel (Std) data, yielding $\Sigma_{\text{L-S}} = \left(\begin{smallmatrix} 111.6 & 0 \\ 0 & 111.6 \end{smallmatrix}\right)$ and $\Sigma_{\text{Std}} = \left(\begin{smallmatrix} 92.7 & 0 \\ 0 & 92.7 \end{smallmatrix}\right)$. In the simulated data, a typical dataset has $\hat{E}(|\mathcal{E}_1|) = 7.3$ m for good GPS data and 14.1 m for bad GPS data (see Section 2.6.1), yielding $\Sigma_{\text{Good}} = \left(\begin{smallmatrix} 84 & 0 \\ 0 & 84 \end{smallmatrix}\right)$, and $\Sigma_{\text{Bad}} = \left(\begin{smallmatrix} 312 & 0 \\ 0 & 312 \end{smallmatrix}\right)$.

The hyperparameters b_1, b_2, s^2, and m_j control the prior distributions on the travel time parameters μ_j and σ_j^2. We set b_1 and b_2 by estimating the possible range in travel time variation for a single link. Some links have very consistent travel times: for example, a link with little traffic and no major intersections at either end. We estimate that such a link could have travel time above or below the median time by a factor of 1.1. Taking this range to be a two standard deviation σ_j interval (so that $1.1 \exp(\mu_j) = \exp(\mu_j + 2\sigma_j)$) yields $\sigma_j \approx 0.0477$. Other links have very variable travel times: for example, a link with substantial traffic. We estimate that such a link could have travel time above or below the median time by a factor of 3.5, corresponding to $\sigma_j \approx 0.6264$. Thus, we set

$b_1 = 0.0477$ and $b_2 = 0.6264$.

We assume there exists an initial travel time estimate τ_j for each link j. For example, in Section 2.7 we use previous estimates from Toronto EMS. We expect this estimate to be typically correct within a factor of two. Thus, we specify m_j and s^2 so that the prior distribution for $E\left(T_{i,j}\right)$ is centered at τ_j and has a two standard deviation interval from $\tau_j/2$ to $2\tau_j$. This gives

$$\begin{aligned}
\tau_j &= E\left(\exp\left(\mu_j + \sigma_j^2/2\right)\right) \\
&= \exp\left(m_j + s^2/2\right) E\left(\exp\left(\sigma_j^2/2\right)\right), \\
\frac{\tau_j}{2} &= \exp\left(m_j + s^2/2 - 2s\right) E\left(\exp\left(\sigma_j^2/2\right)\right), \\
2\tau_j &= \exp\left(m_j + s^2/2 + 2s\right) E\left(\exp\left(\sigma_j^2/2\right)\right),
\end{aligned}$$

where the final equation is redundant. Therefore,

$$m_j = \log\left(\frac{\tau_j}{E\left(\exp\left(\sigma_j^2/2\right)\right)}\right) - \frac{s^2}{2}, \qquad s = \frac{\log(2)}{2}.$$

When τ_j is not available, as in Section 2.6, the following data-based choice for τ_j can be used: find the harmonic mean GPS speed reading in the entire dataset and convert this speed to a travel time for each road.

Results are very insensitive to the hyperparameters b_3 and b_4, as long as the interval $[b_3, b_4]$ does not exclude regions of high likelihood. This is because the entire dataset is used to estimate ζ^2, unlike for the parameters σ_j^2. We fix $b_3 = 0$ and $b_4 = 0.5$. For observed GPS speed V_i^ℓ, suppose the true speed at that moment is v. By Equation 2.3, $V_i^\ell \sim \mathcal{LN}\left(\log(v) - \zeta^2/2, \zeta^2\right)$. If $\zeta = 0.5$, we estimate by simulation that

$$\frac{E\left(\left|V_i^\ell - v\right|\right)}{v} \approx 0.4,$$

which is much higher than any mean absolute error observed by Witte and Wilson [65]. It is not realistic that the speed error could be greater than this.

The constant C governs the multinomial logit choice model on the path traveled. While the results of the IL method are generally insensitive to moderate changes in the other constants, changes in the value of C do have a noticeable effect, so we obtain a careful data-based estimate. Equation 2.1 implies that the ratio of the probabilities of two possible paths depends on their difference in expected travel time. For example, let $C = 0.1$ and consider paths $\tilde{a}_i$ and $\dot{a}_i$ from d_i^s to d_i^f, where the expected travel time of $\tilde{a}_i$ is 10 seconds less than the expected travel time of $\dot{a}_i$. Then path $\tilde{a}_i$ is $e \approx 2.72$ times more likely.

We specify C by the principle that for a trip of average travel time, a driver is ten times less likely to choose a path that has 10% longer travel time. If $\bar{T}$ is the average travel time, then by Equation 2.1, this requires

$$0.1 = \frac{\exp\left(-C\left(1.1\bar{T}\right)\right)}{\exp\left(-C\bar{T}\right)} = \exp\left(-0.1C\bar{T}\right), \tag{2.6}$$

giving $C = -\log(0.1)/\left(0.1\bar{T}\right)$. For our simulated data, $C_{\text{Sim}} = 0.24$.

On the real Toronto data of Section 2.7, we make a small adjustment to pool information across the lights-and-sirens and standard travel datasets. Observing that the route choices are very similar in visual inspection of these datasets, we ensure that the prior distribution on the route taken between two fixed locations is the same for the L-S and Std datasets. To do this, we combine all the L-S and Std data to calculate an overall mean L_1 trip length L_1^{Tor} (change in x coordinate plus change in y coordinate) for the Toronto EMS data, which is $L_1^{\text{Tor}} = 1378.8$m. Let $L_1^{\mathcal{D}}$ and $T^{\mathcal{D}}$ be the mean L_1 length and mean trip time for each dataset $\mathcal{D}$. We estimate a weighted mean time $T_W^{\mathcal{D}} = T^{\mathcal{D}} L_1^{\text{Tor}} / L_1^{\mathcal{D}}$ for dataset $\mathcal{D}$ for a trip of length L_1^{Tor}, and use the time $T_W^{\mathcal{D}}$ to set C by Equation 2.6. This yields $C_{\text{L-S}} = 0.211$ and $C_{\text{Std}} = 0.110$.

2.3.7 Reversibility of the Path Update

The path $A_i = (A_{i,1}, \ldots, A_{i,N_i})$ takes values in the finite set $\mathcal{P}_i$. Conditional on A_i, the vector T_i takes values on the simplex

$$\mathcal{X}_{N_i} \triangleq \left\{ T_i \in \mathbb{R}^{N_i} : T_{i,j} > 0, \sum_{j=1}^{N_i} T_{i,j} = t_i^f - t_i^s \right\},$$

where $t_i^f - t_i^s$ is the known total travel time of trip i. For the reference measure on $\mathcal{X}_{N_i}$ we use $(N_i - 1)$-dimensional Lebesgue measure on the first $N_i - 1$ elements of the vector. Then

$$(A_i, T_i) \in \mathcal{C} \triangleq \bigcup_{A \in \mathcal{P}_i} \{A\} \times \mathcal{X}_{\text{len}(A)}$$

where len(A) is the number of links in $A \in \mathcal{P}_i$. We claim that the move for (A_i, T_i) is reversible with respect to the conditional posterior density of (A_i, T_i) given the GPS data $G = \{G_{i'}\}_{i'=1}^{I}$, the parameters, and the paths and travel times $A_{[-i]}, T_{[-i]}$ for all other trips:

$$\begin{aligned} \nu(A_i, T_i) &\triangleq \pi\left(A_i, T_i \;\middle|\; G, A_{[-i]}, T_{[-i]}, \left\{\mu_j, \sigma_j^2\right\}_{j=1}^{J}, \zeta^2\right) \\ &\propto f_i\left(A_i, T_i, G_i \;\middle|\; \left\{\mu_j, \sigma_j^2\right\}_{j=1}^{J}, \zeta^2\right). \end{aligned} \tag{2.7}$$

Since the dimension of the unknown vector T_i depends on A_i, we treat this as a case of model uncertainty as in Green [21], where the model index k corresponds to the value of $A_i \in \mathcal{P}_i$. Our context, which has an uncertain route for each trip, is slightly different from the context of Green [21], which has a single uncertain model index k and corresponding parameter vector $\theta^{(k)}$. However, Green's argument can still be used to show reversibility of a move for (A_i, T_i) conditional on $A_{[-i]}, T_{[-i]}$ and the parameters $\{\mu_j, \sigma_j^2\}_{j=1}^{J}, \zeta^2$.

Conditional on $A_i^{(1)}$ and $A_i^{(2)}$, we show that our move from $T_i^{(1)} \in \mathcal{X}_{\text{len}\left(A_i^{(1)}\right)}$

to $T_i^{(2)} \in \mathcal{X}_{\text{len}\left(A_i^{(2)}\right)}$ satisfies the dimension-matching condition of Green [21], Section 3.3. We need a bijection between an augmented vector $\left(T_i^{(1)}, u^{(1)}\right)$ and the corresponding augmented vector $\left(T_i^{(2)}, u^{(2)}\right)$, for some $u^{(1)}$ and $u^{(2)}$. Take $u^{(1)} \triangleq \left(T_i^{(2)}(p_1), \ldots, T_i^{(2)}(p_n)\right)$ and $u^{(2)} \triangleq \left(T_i^{(1)}(c_1), \ldots, T_i^{(1)}(c_m)\right)$ and recall that $u^{(1)}$ is drawn independently of $T_i^{(1)}$. Define the bijection $h\left(T_i^{(1)}, u^{(1)}\right) \triangleq \left(T_i^{(2)}, u^{(2)}\right)$ that simply rearranges the elements of the vector $\left(T_i^{(1)}, u^{(1)}\right)$. The absolute value of the Jacobian of such a transformation is one, because that of the identity transform is one, and rearranging the elements corresponds to permuting the rows of the Jacobian, which only changes the sign of the determinant. Although for notational convenience we have included the redundant final elements of the vectors $u^{(1)}$, $u^{(2)}$, $T_i^{(1)}$, and $T_i^{(2)}$, the dimension-matching is on the non-redundant elements of the vectors; in the notation of Green [21], $n_1 = N_i^{(1)} - 1$, $m_1 = n - 1$, $n_2 = N_i^{(2)} - 1$, and $m_2 = m - 1$.

For a dimension-matching move, the acceptance probability that ensures reversibility with respect to a density $\nu(A_i, T_i)$ is given by Equation 7 of Green [21]. It is equal to the absolute value of the Jacobian, times $\frac{\nu\left(A_i^{(2)}, T_i^{(2)}\right)}{\nu\left(A_i^{(1)}, T_i^{(1)}\right)}$, times the ratio of the proposal density of the reverse move relative to that of the proposed move. The probability of proposing a move to $A_i^{(2)}$, given that the current state is $\left(A_i^{(1)}, T_i^{(1)}\right)$, is $\frac{1}{N_i^{(1)} \min\{a^{(1)}, K\}}$ divided by the number of paths of length $\leq K$ from d' to d''. The probability of attempting the reverse move is $\frac{1}{N_i^{(2)} \min\{a^{(2)}, K\}}$ divided by the number of paths of length $\leq K$ from d' to d''. We propose $T_i^{(2)}$ by drawing the subvector $T_i^{(2)}(j) : j \in \{p_1, \ldots, p_n\}$ according to the density

$$\frac{1}{S_i^{n-1}} \text{ Dir}\left(\frac{T_i^{(2)}(p_1)}{S_i}, \ldots, \frac{T_i^{(2)}(p_n)}{S_i}; \alpha\theta(p_1), \ldots, \alpha\theta(p_n)\right)$$

on the simplex $\{T_i \in \mathbb{R}^n : T_{i,j} > 0, \sum_{j=1}^n T_{i,j} = S_i\}$, with respect to $(n-1)$-dimensional Lebesgue measure. The reverse move, from $T_i^{(2)} \in \mathcal{X}_{\text{len}\left(A_i^{(2)}\right)}$ to

$T_i^{(1)} \in \mathcal{X}_{\text{len}\left(A_i^{(1)}\right)}$, proposes $T_i^{(1)}$ by drawing the subvector $T_i^{(1)}(j) : j \in \{c_1, \ldots, c_m\}$ according to the density

$$\frac{1}{S_i^{m-1}} \text{ Dir} \left(\frac{T_i^{(1)}(c_1)}{S_i}, \ldots, \frac{T_i^{(1)}(c_m)}{S_i}; \alpha\theta(c_1), \ldots, \alpha\theta(c_m) \right).$$

Plugging these quantities into Equation 7 of Green [21] and using our Equation 2.7 gives the acceptance probability in our Equation 2.5.

2.4 Comparison Methods

2.4.1 Local Methods

Here we detail the two local methods outlined in Section 2.1. Each GPS reading is mapped to the nearest link (both directions of travel are treated together). Let n_j be the number of GPS points mapped to link j, L_j the length of link j, and $\left\{V_j^k\right\}_{k=1}^{n_j}$ the mapped speed observations. We assume constant speed on each link, as in the IL method. Thus, let $T_j^k = L_j/V_j^k$ be the travel time associated with observed speed V_j^k.

In the first local method, we calculate the harmonic mean of the speeds $\left\{V_j^k\right\}_{k=1}^{n_j}$, and convert to a travel time point estimate

$$\hat{T}_j^H = \frac{L_j}{n_j} \sum_{k=1}^{n_j} \frac{1}{V_j^k}.$$

This is equivalent to calculating the arithmetic mean of the associated travel times T_j^k. The empirical distribution of the associated times $\left\{T_j^k\right\}_{k=1}^{n_j}$ can be used as a distribution estimate. Because readings with speed 0 occur in the Toronto EMS dataset, we set any reading with speed below 5 miles per hour (mph) equal

to 5 mph. This harmonic mean estimator is well-known in the transportation research literature, where it is called the "space mean speed," in the context of estimating travel times using speed data recorded by loop detectors [47, 53, 58].

In Section 2.4.2, we consider this travel time estimator $\hat{T}_j^H$ and its relation to the GPS sampling scheme. We show that if GPS points are sampled by distance (for example, every 200 m), $\hat{T}_j^H$ is an unbiased estimator for the true mean travel time. However, if GPS points are sampled by time (for example, every 10 s), $\hat{T}_j^H$ overestimates the mean travel time. The Toronto EMS dataset uses a combination of sampling-by-distance and sampling-by-time. However, the distance constraint is usually satisfied first (see Figure 2.5, where the sampled GPS points are regularly spaced). Thus, the travel time estimator $\hat{T}_j^H$ is appropriate.

In the second local method, we assume $V_j^k \sim \mathcal{LN}(m_j, s_j^2)$, independently across k, for unknown travel time parameters m_j and s_j^2. This distribution on the travel speed implies that the travel times also have a lognormal distribution: $T_j^k \sim \mathcal{LN}\left(\log(L_j) - m_j, s_j^2\right)$. We use the maximum likelihood estimators (MLEs)

$$\hat{m}_j = \frac{1}{n_j}\sum_{k=1}^{n_j} \log\left(V_j^k\right), \qquad \hat{s}_j^2 = \frac{1}{n_j}\sum_{k=1}^{n_j} \left(\log\left(V_j^k\right) - \hat{m}_j\right)^2$$

to estimate m_j and s_j^2. Our point travel time estimator is

$$\hat{T}_j^{\text{MLE}} = E\left(T_j \middle| \hat{m}_j, \hat{s}_j^2\right) = \exp\left(\log(L_j) - \hat{m}_j + \frac{\hat{s}_j^2}{2}\right).$$

This second local method also provides a natural distribution estimate for the travel times via the estimated lognormal distribution for T_j^k. Correcting for zero-speed readings is again done by thresholding, to avoid taking $\log(0)$.

Some small residential links have no assigned GPS points in the Toronto EMS dataset (see Figure 2.1). In this case, we use a breadth-first search [42] to

find the closest link in the same road class that has assigned GPS points. The road classes are described in Section 2.6; by restricting our search to links of the same class we ensure that the speeds are comparable.

2.4.2 Harmonic Mean Speed and GPS Sampling

When estimating link travel times via speed data from GPS readings, as in the local methods of Section 2.4.1, it is critical whether the GPS readings are sampled by distance (e.g. every 200 m) or by time (e.g. every 10 s). As discussed in Sections 2.1 and 2.4.1, most EMS providers use a combination of distance and time sampling. If both constraints are satisfied frequently, this could create a problem for estimating travel times via these speeds.

In the transportation research literature, speeds are typically recorded by loop detectors at fixed locations on the road, which means that sampling is done by distance. In this context, it is well known that the harmonic mean of the observed speeds (the "space mean speed") is appropriate for estimating travel times [47, 53, 58]. Under a simple probabilistic model of sampling-by-distance, without assuming constant speed, we confirm that the harmonic mean speed gives an unbiased estimator of the mean travel time. However, we also show that if the sampling is done by time, the harmonic mean is biased towards overestimating the mean travel time.

Consider a set of n ambulance trips on a single link. For convenience, let the length of the link be 1. Let the travel time on the link for ambulance i be T_i, and assume that the T_i are iid with finite expectation. Let $x_i(t)$ be the position function of ambulance i, conditional on T_i, so $x_i(0) = 0$ and $x_i(T_i) = 1$. Assume

that $x_i(t)$ is continuously differentiable, with derivative $v_i(t)$, the velocity function, and that $v_i(t) > 0$ for all t. Each trip samples one GPS point. Let V_i^o be the observed GPS speed for the ith ambulance.

First, consider sampling-by-distance. For trip i, draw a random location $\xi_i \sim \text{Unif}(0,1)$ at which to sample the GPS point. This is different from the example of sampling-by-distance above. However, if the sampling locations are not random, we cannot say anything about the observed speeds in general (the ambulances might briefly speed up where the reading is observed, for example). Assuming that the ambulance trip started before this link, it is reasonable to model sampling-by-distance with a uniform random location.

Conditional on T_i, $x_i(\cdot)$ is a cumulative distribution function, with support $[0, T_i]$, density $v_i(\cdot)$, and inverse $x_i^{-1}(\cdot)$. Thus, $\tau_i = x_i^{-1}(\xi_i)$, the random time of the GPS reading, has distribution function $x_i(\cdot)$ and density $v_i(\cdot)$, by the probability integral transform. The observed speed $V_i^o = v_i(\tau_i)$, so the GPS reading is more likely to be sampled when the ambulance has high speed than when it has low speed. This is called the inspection paradox (see e.g. Stein and Dattero [54]). Mathematically,

$$E(V_i^o | T_i) = E(v_i(\tau_i) | T_i) = \int_0^{T_i} v_i(t) v_i(t) dt \geq \frac{\left(\int_0^{T_i} v_i(t) dt\right)^2}{\int_0^{T_i} 1^2 dt} = \frac{1}{T_i},$$

by the Cauchy-Schwarz inequality, with strict inequality unless $v_i(\cdot)$ is constant. However, if we draw a uniform time $\phi_i \sim U(0, T_i)$, then

$$E(v_i(\phi_i) | T_i) = \int_0^{T_i} v_i(t) \frac{1}{T_i} dt = \frac{1}{T_i}. \tag{2.8}$$

The inspection paradox has a greater impact in the Toronto Std data than in the L-S data, because ambulance speed varies more in standard travel.

Consider estimating the mean travel time $E(T_i)$ via the estimator $\hat{T}^H = 1/\bar{V}_H^o$, where $\bar{V}_H^o$ is the harmonic mean observed speed. We have

$$E\left(\hat{T}^H\right) = E\left(E\left(\hat{T}^H \middle| \{T_i\}_{i=1}^n\right)\right) = E\left(\frac{1}{n}\sum_{i=1}^n E\left(\frac{1}{v_i(\tau_i)}\middle| T_i\right)\right)$$
$$= E\left(\frac{1}{n}\sum_{i=1}^n \int_{t=0}^{T_i} \frac{1}{v_i(t)} v_i(t)dt\right) = E\left(\frac{1}{n}\sum_{i=1}^n T_i\right) = E(T_i),$$

and so it is unbiased.

Next, suppose the sampling is instead done by time. To model this, let $\tau_i \sim$ Unif$(0, T_i)$ be a random time to sample the GPS point for ambulance i. In this case, we have

$$E\left(\hat{T}^H\right) = E\left(\frac{1}{n}\sum_{i=1}^n E\left(\frac{1}{v_i(\tau_i)}\middle| T_i\right)\right)$$
$$\geq E\left(\frac{1}{n}\sum_{i=1}^n \frac{1}{E\left(v_i(\tau_i)\middle| T_i\right)}\right)$$
$$= E\left(\frac{1}{n}\sum_{i=1}^n \frac{1}{\frac{1}{T_i}}\right) = E(T_i),$$

by Jensen's Inequality and Equation 2.8. Again, the inequality is strict unless $v_i(\cdot)$ is constant.

2.4.3 Method of Budge et al.

Budge, Ingolfsson and Zerom [8] introduced a travel time distribution estimation method relying on trip distance. Since the exact path traveled is usually unknown, the length of the shortest distance path between the start and end locations is used as a surrogate for the true travel distance. The method relies on the model $t_i = m(d_i)\exp\left[c(d_i)\epsilon_i\right]$, where t_i and d_i are the total time and distance for trip i, ϵ_i follows a t-distribution with τ degrees of freedom, and $m(\cdot)$ and

$c(\cdot)$ are unknown functions. In their preferred method, they assume parametric expressions for the functions $m(\cdot)$ and $c(\cdot)$, and estimate the parameters using maximum likelihood.

We implemented this parametric method and compared it to a related binning method. In the binning method, we divide the ambulance trips into bins by trip distance, and fit a separate t-distribution to the log travel times for each bin. We then linearly interpolate between the quantiles of the travel time distributions for adjacent bins, to generate a travel time distribution estimate for a trip of any distance. On simulated data on the Toronto subregion, the parametric and binning methods perform very similarly, while on real data on the subregion, the binning method slightly outperforms the parametric method. Thus, we report only results of the binning method in Sections 2.6-2.7.

2.5 Bias Correction

We use a bias correction factor to make each method approximately unbiased, because we have found that this improves performance for all methods. There are several reasons why the methods result in biased estimates, some inherent to the methods themselves and some due to sampling characteristics of the GPS data. One source of bias is the inspection paradox in the GPS data, discussed in Section 2.4.2. The IL method is also biased because of the difference in path estimation from the training to the test data. On the training data, the IL method uses the GPS data to estimate a solution to the map-matching problem. On the test data, the estimated fastest path between the start and end nodes is used, to imitate the prediction scenario where the route is not known beforehand. This

leads to underestimation of the true travel times.

Most commonly, bias correction is done using an asymptotic expression for the bias [6, 29]. We use an empirical bias correction factor, because there is no analytic expression available. The bias correction factor for each method is calculated in the following manner. We divide the set of trips from each dataset randomly into training, validation, and test sets [22]. We fit the methods on the training data, calculate a bias correction factor on the validation data, and predict the travel times for the trips in the test data. The data are split into 50% training and 50% validation and test. To use the validation/test data most efficiently, we do cross-validation: divide the validation/test data into ten sets, use nine sets for the validation data, the tenth for the test data, and repeat for all ten cases. For a given validation set of n trips, where the estimated trip travel times are $\{\hat{t}_i\}_{i=1}^n$ and the true travel times are $\{t_i\}_{i=1}^n$, the bias correction factor is

$$b = \frac{1}{n}\left(\sum_{i=1}^{n} \log \hat{t}_i - \sum_{i=1}^{n} \log t_i\right)$$

Subtracting this factor from the log estimates on the test data makes each method unbiased on the log scale. We calculate the bias correction on the log scale because it is more robust to travel time outliers.

2.6 Simulation Experiments

Next we test the IL method, local methods, and the method of Budge et al. on simulated data. We compare the accuracy of the four methods for predicting travel times of test trips. We simulate ambulance trips on the road network of Leaside, Toronto, shown in Figure 2.1 (roughly 4 square kilometers). This region has four road classes; we define the highest-speed class to be primary links, the

two intermediate classes to be secondary links, and the lowest-speed class to be tertiary links (Figure 2.1). In the Leaside region, a value $K = 6$ (see Section 2.3.5) guarantees that the Markov chain is valid.

2.6.1 Generating Simulated Data

We simulate ambulance trips with true paths, travel times, and GPS readings. For each trip i, we uniformly choose start and end nodes. We construct the true path A_i link-by-link. Beginning at the start node, we uniformly choose an adjacent link from those that lower the expected time to the end node, and repeat until the end node is reached. This method differs from our Bayesian prior (see Section 2.2.1), and can lead to a wide variety of paths traveled between two nodes.

The link travel times are lognormal: $T_{i,k} \sim \mathcal{LN}(\mu_{A_{i,k}}, \sigma^2_{A_{i,k}})$. To set the true travel time parameters $\left\{\mu_j, \sigma_j^2\right\}$ for link j, we uniformly generate a speed between 20-40 mph. We draw $\sigma_j \sim$ Unif $\left(0.5 \log\left(\sqrt{3}\right), 0.5 \log(3)\right)$, and set μ_j so that the link length divided by the mean travel time equals the random speed. The range for σ_j generates a wide variety of link travel time variances. Comparisons between the estimation methods are invariant to moderate changes in the σ_j range.

We simulate datasets with two types of GPS data: good and bad. The good GPS datasets are designed to mimic the conditions of the Toronto EMS dataset. Each GPS point is sampled at a travel distance of 250 m after the previous point. Straight-line distance between GPS readings is typically 200 m in the Toronto EMS data, but we simulate data via the longer along-path distance. The GPS

locations are drawn from a bivariate normal distribution with $\Sigma = \left(\begin{smallmatrix} 100 & 0 \\ 0 & 100 \end{smallmatrix}\right)$. The GPS speeds are drawn from a lognormal distribution with $\zeta^2 = 0.004$, which gives a mean absolute error of 5% of speed, approximately the average result seen by Witte and Wilson [65].

The bad GPS datasets are designed to be sparse and have GPS error consistent with the high error results seen by Chen et al. [9] and Witte and Wilson [65]. GPS points are sampled every 1000 m. The constant $\Sigma = \left(\begin{smallmatrix} 465 & 0 \\ 0 & 465 \end{smallmatrix}\right)$, which gives mean distance of 27 m between the true and observed locations, the average error seen in Hong Kong by Chen et al. [9]. The parameter $\zeta^2 = 0.01575$, corresponding to mean absolute error of 10% of speed, which is approximately the result from low-quality GPS settings tested by Witte and Wilson [65].

2.6.2 Travel Time Prediction

We simulate ten good GPS datasets and ten bad GPS datasets, as defined above, each with a training set of 2000 trips and a validation/test set of 2000 trips. Taking the true path for each test trip as known and using the cross-validation approach of Section 2.5 to estimate bias correction factors, we calculate point and 95% predictive interval estimates for the test set travel times using the four methods. To obtain a gold standard for performance, we implement an Oracle method. In this method, the true travel time parameters $\left\{\mu_j, \sigma_j^2\right\}$ for each link j are known. The true expected travel time for each test trip is used as a point estimate. This implies that the Oracle method has the lowest possible root mean squared error (RMSE) for realized travel time estimation.

We compare the predictive accuracy of the point estimates from the four

methods via the RMSE (in seconds), the RMSE of the log predictions relative to the true log times ("RMSE log"), and the mean absolute bias on the log scale over the test sets of the cross-validation procedure ("Bias M.A."). We calculate metrics on the log scale because the residuals on the log scale are much closer to normally distributed. On the original scale, there are several outlying trips in the Toronto EMS data (Section 2.7) with very large travel times that heavily influence the metrics. The bias metric measures how well the bias correction works. If $k \in \{1, \ldots, 10\}$ indexes the cross-validation test sets, where test set k has n_k trips with true travel times $t_i^{(k)}$ and estimates $\hat{t}_i^{(k)}$, for $i \in \{1, \ldots, n_k\}$, then

$$\text{Bias (M.A.)} = \frac{1}{10} \sum_{k=1}^{10} \left| \frac{1}{n_k} \left(\sum_{i=1}^{n_k} \log \hat{t}_i^{(k)} - \sum_{i=1}^{n_k} \log t_i^{(k)} \right) \right|. \tag{2.9}$$

We compare the interval estimates using the percentage of 95% predictive intervals that contain the true travel time ("Cov. %") and the geometric mean width of the 95% predictive intervals ("Width"). Table 2.1 gives arithmetic means for these metrics over the ten good and bad simulated datasets.

In both dataset types, the point estimates from the IL method greatly outperform the estimates from the local methods and the method of Budge et al. The IL estimates closely approach the Oracle estimates, especially on the good GPS datasets. In the good datasets, the IL method has 70% lower error than the local methods in RMSE on the log scale, and 78% lower error than Budge et al., after eliminating the unavoidable error of the Oracle method. In the bad datasets, the IL method outperforms the local methods by 70% and Budge et al. by 56% in log scale RMSE, relative to the Oracle method. The method of Budge et al. outperforms the local methods on the bad GPS data, while the reverse holds for the good GPS data.

Good GPS data (Mean over ten datasets)					
Estimation method	RMSE (s)	RMSE log	Bias (M.A.)	Cov. %	Width (s)
Oracle	15.9	0.183	0.010	-	-
IL	16.1	0.187	0.010	95.8	57.2
Local MLE	16.8	0.196	0.010	94.4	56.8
Local Harm.	16.8	0.196	0.010	94.0	56.2
Budge et al.	17.3	0.201	0.011	96.2	67.2
Bad GPS data (Mean over ten datasets)					
Estimation method	RMSE (s)	RMSE log	Bias (M.A.)	Cov. %	Width (s)
Oracle	16.4	0.183	0.012	-	-
IL	16.9	0.191	0.013	96.1	60.4
Local MLE	18.1	0.209	0.014	92.3	57.8
Local Harm.	18.1	0.209	0.014	90.9	55.5
Budge et al.	17.9	0.201	0.013	96.2	68.2

Table 2.1: Out-of-sample trip travel time estimation performance on simulated data.

The IL method also outperforms the other methods in interval estimates. For the good GPS data, the interval estimates from the IL and local methods are similar, while the estimates from the method of Budge et al. are substantially wider, with slightly higher coverage percentage. For the bad GPS data, the intervals from the IL method have higher coverage percentage than the intervals from the local methods, and the intervals from the method of Budge et al. are again wider, with no corresponding increase in coverage percentage.

2.6.3 Map-Matching Results

Next we assess path estimates from the IL method for representative paths, shown in Figure 2.2. The GPS locations are shown in white. The starting node is marked with a cross and the ending node with an X. Each link is shaded in gray by the marginal posterior probability that it is traversed in the path. Links with

probability less than 1% are unshaded. The left-hand path is from a good GPS dataset, as defined in Section 2.6.1. The IL method easily identifies the correct path. Every correct link has close to 100% probability, and only two incorrect detours have probability above 1%. This is typical performance for trips with good GPS data. The right-hand path is from a bad GPS dataset. The sparsity in GPS readings makes the path very uncertain. Near the beginning of the path, there are five routes with similar expected travel times, and the GPS readings do not distinguish between them, so each has roughly 20% posterior probability. The IL method is very effective at identifying alternative routes when the true path is unclear.

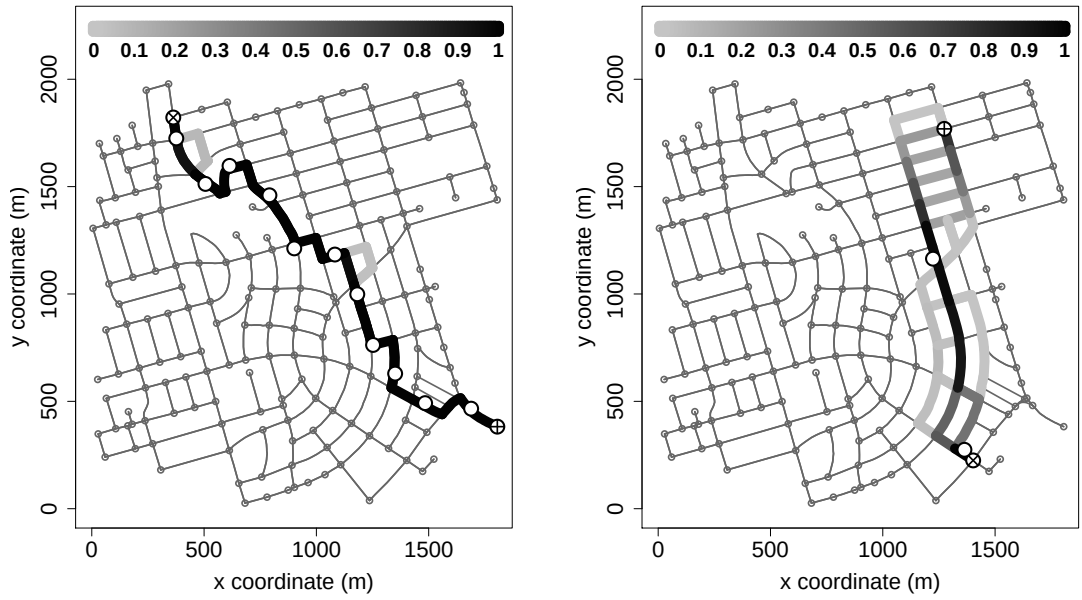

Figure 2.2: Map-matching estimates for two simulated trips, shaded by the probability each link is traversed.

2.7 Analysis of Toronto EMS Data

Next we compare the IL method and alternative methods on the Toronto data.

2.7.1 Data

The Toronto EMS data consist of GPS data and trip information for ambulance trips with one of two priority levels: lights-and-sirens (L-S) or standard travel (Std). We address these separately, again focusing on the Leaside subregion of Toronto. The right plot in Figure 2.1 shows the GPS locations for the L-S dataset. This dataset contains 1930 ambulance trips and roughly 14,000 GPS readings. The primary roads tend to have a large amount of data, the secondary roads a moderate amount, and the tertiary roads a small amount. The Std dataset is larger (3989 trips), with a similar spatial distribution of GPS locations.

We use only the portion of trips where the ambulance was driving to the scene of an emergency, and discard trips for which this portion cannot be identified. We also discard some trips (roughly 1%) that would impair estimation: for example, trips where the ambulance turned around or where the ambulance stopped for a long period, not at a stoplight or in traffic. Finally, most of the trips in the dataset do not begin or end in the subregion, they simply pass through, so we use the closest node to the first GPS location as the approximated start node, and the time of the first GPS reading as the start time. Similarly, we use the last GPS reading for the end node. This produces some inaccuracy of estimated travel times on the boundary of the region. This could be fixed by applying our method to overlapping regions and discarding estimates on the boundary.

2.7.2 Link Travel Time Estimates

Here we report the travel time estimates from the IL method. Toronto EMS has existing estimates of the travel times, which we use to set the prior $\{m_j\}_{j=1}^{J}$ hyperparameters (see Section 2.3.6). These estimates are different for L-S and Std trips, but are the same for the two travel directions of parallel links. We have also tested the IL method with the data-based hyperparameters described in Section 2.3.6 and have observed similar performance. Figure 2.3 shows prior and posterior speed estimates (length divided by mean travel time) from the IL method on the L-S dataset. Each link is shaded in gray based on its speed estimate, so most roads have two shades in the right-hand plot, corresponding to travel in each direction.

The posterior speed estimates from the IL method are reasonable; primary links tend to have high speed estimates, and estimated speeds for consecutive links on the same road are typically similar. Links heading into major intersections (intersections between two primary or secondary roads, as shown in Figure 2.1) are often slower than the reverse links. In the corresponding figure for Std data (not shown), the slowdown into major intersections is even more pronounced. For most links, the posterior estimate of the speed is higher than the prior estimate, suggesting that the existing road speed estimates used to specify the prior are underestimates.

There are a few links that have poor estimates from the IL method. For example, parallel black links in the top-left corner have poor estimates due to edge effects. Also, some short interior links have unrealistically high estimates, likely because there are few GPS points on these links. This undesirable behavior could be reduced or eliminated by using a random effect prior distribution [16]

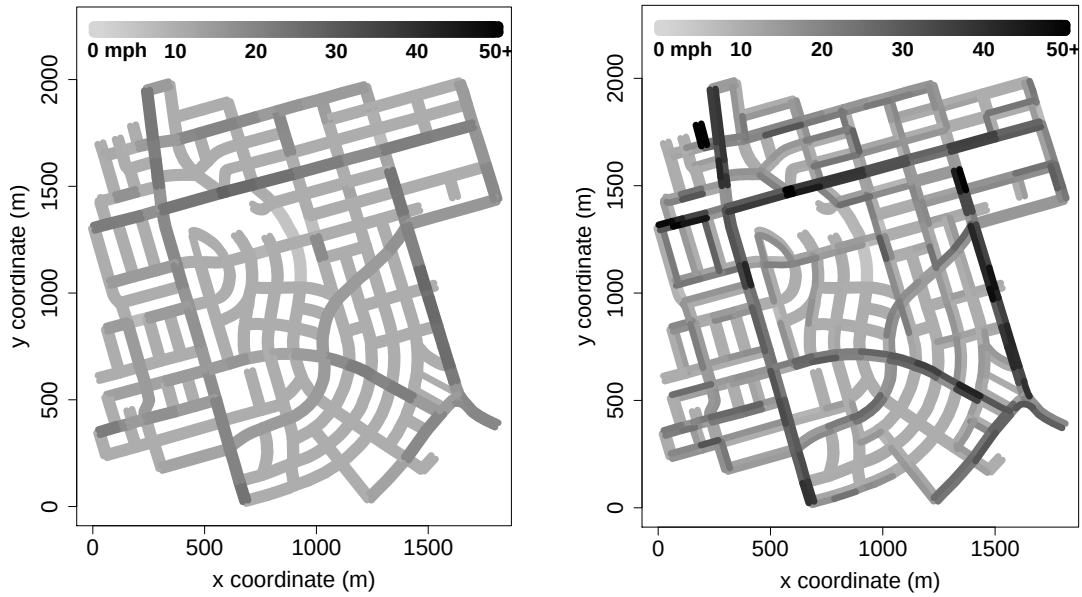

Figure 2.3: Prior (left) and posterior (right) speeds from the IL method, for Toronto L-S data, in miles per hour (mph).

for roads in the same class, which has the effect of pooling the available data.

2.7.3 Travel Time Prediction

We compare the known travel time of each trip in the test data with the point and 95% interval predictions from each method. Unlike the simulated test data in Section 2.6, the true paths are not known. For the IL and local methods, we assume that the path taken is the fastest path in expectation. This measures the ability of each method to estimate both the fastest path and the travel time distributions.

We again use the cross-validation approach of Section 2.5 to estimate bias correction factors. We resample random training and validation/test sets five

times, and give arithmetic means of the performance metrics over the five replications in Table 2.2. We again compare the point estimates from the three methods on the test data using RMSE, RMSE log, and Bias (M.A.), and compare the interval estimates using Width and Cov. %. Because the true travel time distributions are unknown, we cannot use the Oracle method as in Section 2.6.2. However, we still wish to estimate gold standard performance, so we implement an Estimated Oracle method, in which we assume that the parametric model and estimates from the Local MLE method are the truth. We simulate realized travel times on the fastest path (in expectation, as estimated by the Local MLE method) for each test trip, and compare these to the point estimates from the Local MLE method. To avoid simulation error, we use Monte Carlo estimates from 1000 simulated travel times for each trip.

L-S data (Mean over five replications)					
Method	RMSE (s)	RMSE log	Bias (M.A.)	Cov. %	Width (s)
Est. Oracle	14.9	0.168	0.018	-	-
IL	37.8	0.332	0.025	85.8	75.0
Local MLE	38.4	0.342	0.027	73.3	55.0
Local Harm.	38.5	0.343	0.028	77.5	75.2
Budge et al.	39.8	0.342	0.028	94.5	122.3
Std data (Mean over five replications)					
Method	RMSE (s)	RMSE log	Bias (M.A.)	Cov. %	Width (s)
Est. Oracle	35.2	0.191	0.018	-	-
IL	126.8	0.465	0.025	73.0	141.8
Local MLE	129.0	0.480	0.025	58.4	118.6
Local Harm.	129.0	0.480	0.025	64.8	142.8
Budge et al.	127.9	0.475	0.026	94.3	370.8

Table 2.2: Out-of-sample trip travel time estimation performance on Toronto EMS data.

For the L-S data, the IL method outperforms the method of Budge et al. and the local methods, suggesting that it is effectively combining trip information with GPS information. The IL method is roughly 6% better in log scale RMSE,

after subtracting the error from the Estimated Oracle method. The method of Budge et al. and the local methods perform similarly. The bias correction is successful at eliminating bias (there is 2-3% bias remaining).

The IL method substantially outperforms the local methods in interval estimates. The IL method intervals have much higher coverage percentage than the intervals from the local methods. The method of Budge et al. has higher coverage percentage than the IL method; however, the intervals are also wider. The intervals from the MLE method are narrow and have low coverage percentage. Therefore, the Local MLE method does not adequately account for travel time variability, suggesting that the Estimated Oracle method may underestimate the baseline error. If so, the IL method outperforms the other methods by an even larger amount, relative to the baseline error.

For the Std data, the IL method outperforms the local methods by roughly 5% in RMSE on the log scale, and outperforms the method of Budge et al. by 3.5%, again relative to the Estimated Oracle error. Point estimates from the method of Budge et al. slightly outperform the local methods. Interval estimation is less successful for the IL and local methods than for the L-S data, probably because the Std travel times have more unaccounted sources of variability than the L-S travel times, such as traffic and time of day.

This region and dataset are generally favorable to the method of Budge et al. The travel speeds are similar across most roads in this region, which mitigates the main weakness of the Budge et al. method, namely its inability to distinguish between fast and slow roads. Also, several particular paths are very common in the Leaside region, and the Budge et al. method fits the travel time distribution of these particular paths very closely, leading to relatively high predictive

accuracy. On the full city the routes would be much more heterogeneous, with many different routes of roughly the same travel distance, so that a method that can model the heterogeneity is expected to have a greater advantage.

2.7.4 Probability of Arrival Within a Time Threshold

Next we estimate the probability an ambulance completes its trip within a certain time threshold [8]. These probabilities are useful for EMS providers (see Section 2.1). In Figure 2.4, we assume that an ambulance begins at the node marked with a black X and estimate the probability it reaches each other node in 150 seconds, following the fastest path in expectation. For the IL method, these probabilities are calculated by simulating travel times from the posterior distribution of each link in the route, and using Monte Carlo estimation. The left-hand figure shows probabilities from the IL method, and the right-hand figure shows probabilities from the method of Budge et al.

The probabilities for both methods appear reasonable; they are high for nodes close to the start node and decrease for nodes further away. The probabilities from the IL method appear more realistic than those from Budge et al., since nodes on main roads tend to have higher probabilities from the IL method (for example, traveling south from the start node), whereas nodes on minor roads far from the start node have lower probabilities from the IL method (see the bottom-right in each plot). This is because the method of Budge et al. does not take into account the different speeds of different roads.

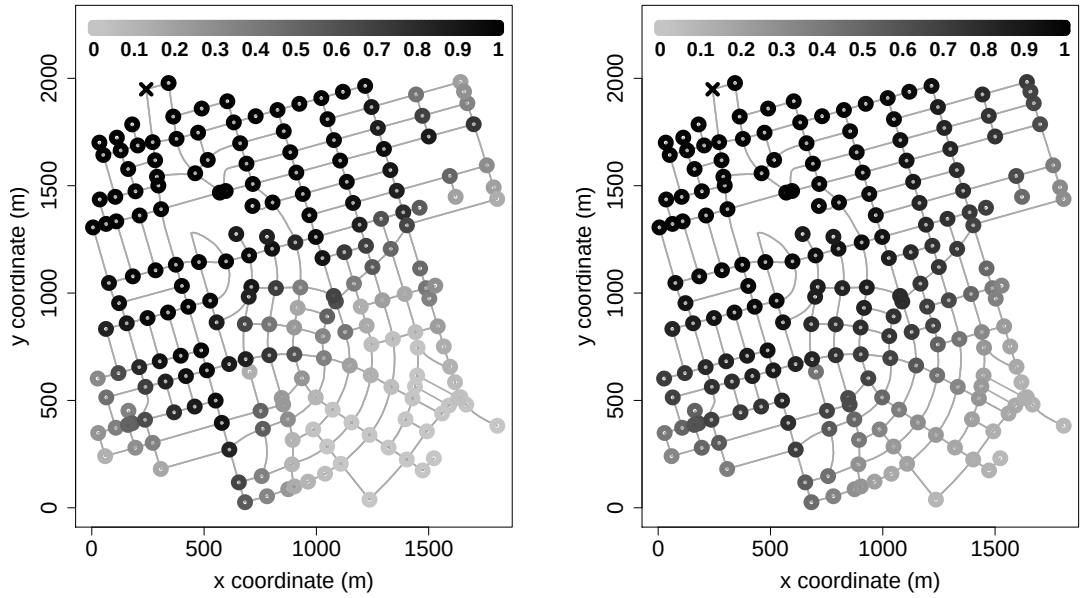

Figure 2.4: Estimates of probability of reaching each node in 150 seconds, IL method (left), Budge et al. method (right), from the location marked X.

2.7.5 Map-Matching Results

Finally, we assess map-matching estimates from the IL method, for the Toronto L-S data. Figure 2.5 shows two example ambulance paths from the L-S dataset. The GPS locations are shown in white; the first reading is marked with a cross and the last with an X. As in Section 2.6.3, each link is shaded by its marginal posterior probability, if it is greater than 1%. In the left-hand path, there are two occasions where the path is not precisely clear from the GPS readings. On both occasions, roughly 90% of the posterior probability is given to a route following the main road, which is estimated to be faster. The final two GPS readings appear to have location error. However, the fastest path is still given roughly 100% posterior probability, instead of a detour that would be slightly closer to the second-to-last GPS reading. In the right-hand path, for an unknown reason,

there is a large gap between GPS points. Most of the posterior probability is given to the fastest route along the main roads. This illustrates the robustness of the IL method to sparse GPS data.

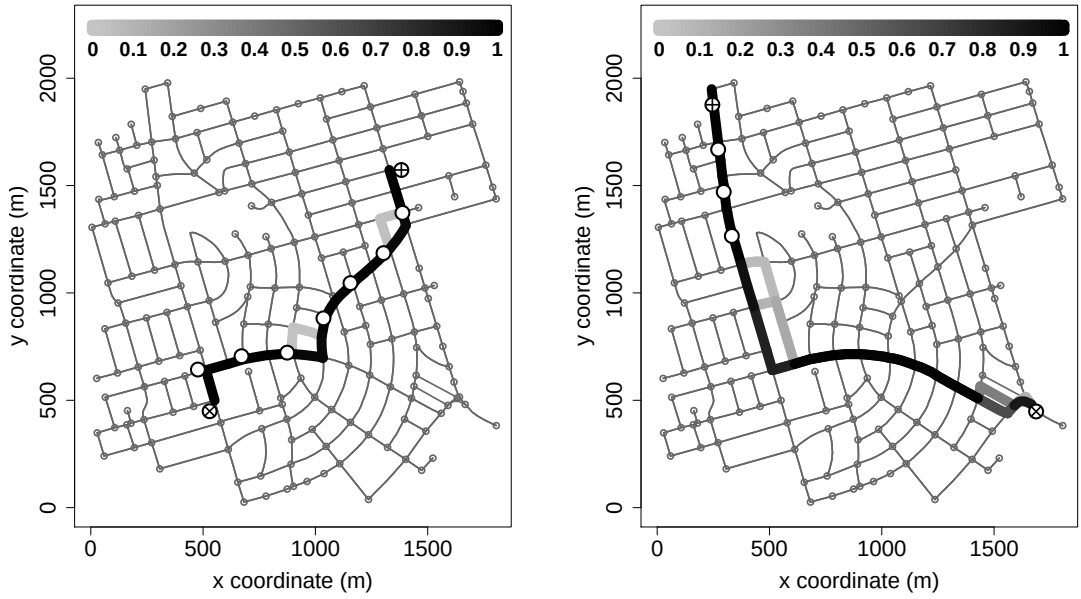

Figure 2.5: Map-matching estimates for two Toronto L-S trips, shaded by the probability each link is traversed.

2.8 Conclusions

We proposed a Bayesian method, called the Independent Link (IL) method, to estimate the travel time distribution on any route in a road network. We simultaneously estimated the vehicle paths and the parameters of the travel time distributions. We also introduced two local methods based on mapping each GPS reading to the nearest link. The first method used the harmonic mean of the GPS speeds; the second performed maximum-likelihood estimation for a parametric distribution of travel speeds on each link.

We compared these three methods to an existing method from Budge et al. [8]. In simulations, the IL method greatly outperformed the local methods and the method of Budge et al. in estimating out-of-sample trip travel times, for both point and interval estimates. The estimates from the IL method remained excellent even when the GPS data had high error. On the Toronto EMS data, the IL method outperformed the competing methods in out-of-sample point estimation, though interval estimates were slightly narrow. The IL method provided more realistic estimates of the probability of completing a trip within a time threshold than the method of Budge et al.

In the next chapter, we consider modifications to the IL model, addressing several issues. First, we include time-varying travel times, because speeds typically decrease during rush hour, for example. Applying the IL method separately to rush hour and non-rush hour improves performance on standard travel Toronto data, but has little effect on performance for lights-and-sirens data. Second, we modify the IL model to obtain more efficient computation on large road networks. Third, we investigate information sharing across roads, to improve estimates on infrequently-used roads. Finally, we incorporate dependence between link travel times within each trip. This change improves coverage of interval estimates.

CHAPTER 3

LARGE-NETWORK TRAVEL TIME DISTRIBUTION ESTIMATION, WITH APPLICATION TO AMBULANCE FLEET MANAGEMENT

3.1 Introduction

Predictions of vehicle travel times are necessary for navigation systems, transport policy decisions, and management of vehicle fleets such as taxi and transit vehicles, emergency vehicles, and delivery services [10]. Travel time predictions are used not only for vehicle routing, but for traffic management, dispatch decisions, and real-time deployment algorithms for emergency vehicles [7, 10, 27]. In many of these applications it is also important to capture the uncertainty in the travel time, by predicting the entire travel time distribution rather than just the expected travel time [48]. For instance, taking into account uncertainty in the travel time of ambulances to the scene of an emergency can substantially increase the survival rate of cardiac patients, by improving fleet management decisions and thus reducing response times [12, 40]. Also, ambulance fleet performance is measured by the fraction of emergency calls for which the response time is less than a certain threshold [36].

We propose a new method for predicting the distribution of a vehicle travel time on an arbitrary route in a road network. The prediction depends on the route and on explanatory variables such as the time of day and day of week. Our method uses information from historical trips on the network, specifically the total travel time and estimated path for each trip. In order to predict the travel time distribution for a particular route, we do not require historical trips that take precisely the same route. Instead, our statistical approach uses infor-

mation from all the historical trips by learning shared properties like the effects of time of day and types of road traversed. The model we use is intuitive and its parameters are interpretable. Our method is computationally efficient, scaling effectively to large road networks and large historical trip databases. It is designed for contexts in which the historical trips are sparse in time, so that incorporation of traffic flow patterns is infeasible. If data are available more densely in time, a method incorporating traffic dynamics may be more effective [24, 25]. Further, our method is most useful in contexts where the historical trips are the most relevant source of information, such as travel time estimation for fleet vehicles, which tend to behave in a consistent manner that can be different from other types of vehicles. We highlight the context of ambulance fleets, describing modeling choices motivated by that context, although our model framework is more generally stated and applicable to other contexts.

The historical trip data used by our method can be obtained from a variety of sources; most importantly, Global Positioning System (GPS) measurements from vehicles traveling on the network can be used to estimate the routes traveled by the vehicles, even if the GPS measurements are recorded infrequently [34, 36, 41, 45, 46]. This source of data is called floating car data or automatic vehicle location data, and is increasingly available for taxi fleets, delivery services, emergency vehicle fleets, and personal vehicles via GPS-enabled smartphones or 2-way navigation devices (e.g. Garmin or TomTom). Unlike other sources of travel time data, it does not require instrumentation on the roadway, and thus is the only source of data available to estimate travel times that has the prospect of comprehensive network coverage [25].

There are still few methods available to utilize floating car data for travel

time distribution prediction. Hofleitner, Herring, and Bayen [25] and Hofleitner et al. [24] take a traffic flow perspective, modeling at the level of the network link (a road segment between two intersections). They use a dynamic Bayesian network for the unobserved traffic conditions on links and model the link travel time distributions conditional on the traffic state. Their method is applied to a subset of the San Francisco road network with roughly 800 links, predicting travel times using taxi fleet data and validating with additional data sources.

In the previous chapter, we introduced our IL method for simultaneous travel time distribution and path estimation for a set of vehicle trips [62]. Like Hofleitner et al., we modeled travel times at the link level. We applied the IL method to estimate ambulance travel times on a subregion of Toronto.

Jenelius and Koutsopoulos [28] propose a framework for estimating the distribution of travel times while incorporating weather, speed limit, and other explanatory factors. They point out that approaches such as Hofleitner et al. and our IL method [24, 25, 62] assume that the link travel times are independent within a vehicle trip, perhaps conditional on the traffic state. This contrasts with empirical evidence suggesting that the link travel times are strongly correlated, even after conditioning on time of day and other explanatory factors [3, 48]. Therefore, they capture correlation using a moving average specification for the link travel times. Their model is applied to estimate travel times for a particular route in Stockholm.

In contrast to these approaches, in this chapter we model travel times at the trip level instead of the link level. This naturally incorporates dependence between link travel times. For this reason, we refer to our method as the Whole Trip (WT) method. The vehicle route is taken into account in the specification of

the trip travel time parameters, such as the median travel time. This trip-level approach is related to that of Budge, Ingolfsson, and Zerom [8], who model the travel time distribution for an ambulance trip as a function of shortest-path distance between the start and end locations. They assume that the log travel time follows a t-distribution, and propose nonparametric and parametric representations of the centering and scale parameters, as functions of the shortest-path distance between start and end locations. Like them we take a regression approach, but we also incorporate dependence on the route taken, time of day, and other explanatory factors, justifying our modeling choices empirically.

We use our WT method to predict ambulance travel times for the entire road network of Toronto. The size of the road network (68,272 links) is an order of magnitude larger than in previous applications of travel time distribution estimation based on floating car data [8, 24, 25, 28, 62], and the number of historical vehicle trips (157,283) is also larger than these previous applications. We compare the prediction accuracy of our WT method to that of Budge et al. [8], our IL method of the previous chapter [62], and a commercial software package for mean travel time estimation. We also consider the effect of various simplifications of our WT model, and investigate the accuracy of our model when the time effect on travel times is artificially inflated.

Finally, we evaluate the effect of using our WT method for ambulance fleet management, relative to that of Budge et al. [8]. We do this by selecting a set of representative ambulance posts in Toronto. We calculate which ambulance post is estimated to be the closest in median travel time to each intersection in Toronto, and find that many intersections have different estimated closest posts, according to the two methods. Therefore, the two methods would rec-

ommend that a different ambulance respond to emergencies at these locations, if the closest ambulance is dispatched. We also calculate the probability that an ambulance is able to respond on time (within a specified time threshold) from the closest post to each intersection of the city. We find substantial differences in these probabilities between the two methods. As in the previous chapter, these appear to arise because our WT method captures differences in speeds between different types of roads, unlike the method of Budge et al.

Commercially-available vehicle travel time estimates typically consist of estimated expected travel times rather than distribution estimates, so they cannot be used for applications that require a travel time distribution, such as ambulance deployment algorithms using simulated travel times. Also, these estimates are calculated for standard vehicle speeds, not "lights-and-sirens" ambulance speeds. However, they are still useful for point estimation performance comparisons, as long as they are corrected for bias. Specifically, we investigate travel time estimates from TomTom, a maker of navigation devices.

This chapter is organized as follows. In Section 3.2, we introduce the WT statistical model and estimation method. In Section 3.3, we introduce the data from Toronto and highlight the exploratory data analysis that motivates our modeling choices. We discuss data preprocessing in Section 3.3.1. In Section 3.4, we give details on the estimates from TomTom. In Section 3.5, we discuss estimation results from the WT method and comparisons with the alternative methods. We draw conclusions in Section 3.6.

3.2 Modeling and Estimation

3.2.1 Travel Time Modeling

Consider a road network with links indexed by $j \in \{1, \ldots, J\}$ and a set of vehicle trips on that network indexed by $i \in \{1, \ldots, I\}$. Let d_j indicate the length of link j. Assume that each trip i begins and ends at known locations on the road network (not necessarily at intersections), and that the sequence of links $A_i = \{A_i^1, \ldots, A_i^{n_i}\}$ traversed by trip i is known. Let f_{ij} denote the known fraction of link j used by trip i. For interior links in the path A_i, this fraction equals 1; for the first and last links, it captures the fraction of the link actually traversed during the trip.

In our WT method, the travel time T_i for trip i is modeled with a lognormal distribution, conditional on the route traveled. Specifically,

$$T_i \,|A_i, \{f_{ij}\}_{j\in A_i}, \{d_j\}_{j\in A_i} \sim \mathcal{LN}\left(\mu(i) + \log\left(c + \sum_{j\in A_i} f_{ij} d_j u(i,j)\right), \sigma^2(i)\right) \quad (3.1)$$

conditionally independent across trips i, where the functional forms of $\mu(i)$, $u(i,j)$, and $\sigma^2(i)$ are specified appropriately for the context. This model can be rewritten as $T_i = R_i(c + \sum_{j\in A_i} f_{ij} d_j u(i,j))$ for a random lognormal multiplicative factor $R_i \sim \mathcal{LN}(\mu(i), \sigma^2(i))$ capturing the travel time variability and trip-level effects. The baseline travel time is given by $c + \sum_{j\in A_i} f_{ij} d_j u(i,j)$, where the term $u(i,j)$ is a unit travel time (inverse of speed) for trip i on link j. The product $f_{ij} d_j$ is the distance traveled on link j in trip i, so the baseline travel time is a sum of individual link travel times plus an intercept $c > 0$. Intersection and turn effects can also be included in the specification; we do not focus on this extension because it has a minor effect on predictive accuracy in our application

to ambulance travel times, since ambulances do not have to obey traffic signals when traveling at lights-and-sirens speeds.

The intercept c captures, for instance, additional time required to get up to speed at the beginning of the trip and to slow down at the end. Its inclusion is similar to the model introduced by Kolesar et al. [32] and used by Budge et al. [8], in which the travel times depend on the square root of the distance for small distances, and grow linearly with the distance for large distances. If the linear part of this model is extrapolated to distance 0, the intercept is positive.

The unit travel time $u(i, j)$ for link j in trip i can depend on explanatory factors like the road class, speed limit, and whether the road is one-way. Most simply it can be a link effect, giving the form $u(i, j) \triangleq u_j$. However, if there are links with very few trips, as is the case for ambulance data, this approach yields noisy estimates of the u_j parameters. For the ambulance study, we specify $u(i, j)$ to depend on the road class, taking $u(i, j) \triangleq u_{\ell(j)}$ where $\ell(j) \in \{1, \ldots, L\}$ is the road class of link j (highway, arterial road, etc.). Alternatively, the road network could be partitioned into R geographic regions, using $u(i, j) \triangleq u_{\ell(j),r(j)}$ for $r(j) \in \{1, \ldots, R\}$, to allow downtown arterial roads to be distinguished from suburban arterial roads, for example.

The parameters $\mu(i)$ and $\sigma^2(i)$ for the trip effect can depend on time, weather, vehicle type, driver, and other explanatory factors (similarly to Jenelius et al. [28]). For the ambulance study, we use time bin as an explanatory factor, setting $\mu(i) \triangleq \mu_{k(i)}$ where the week is divided into time bins $k \in \{0, 1, \ldots, K\}$ and $\mu_0 \triangleq 0$ to ensure model identifiability, i.e. to ensure that each parameter of the model can be uniquely determined from sufficient data.

For the ambulance study, we specify the log-scale variance $\sigma^2(i)$ using an exponential decay model in the total trip distance $d_i \triangleq \sum_{j \in A_i} f_{ij} d_j$, as suggested by exploratory data analysis (Section 3.3.2). Specifically, we take $\sigma^2(i) \triangleq Me^{-\lambda d_i} + \delta$, for parameters $M > 0$, $\lambda > 0$, and $\delta > 0$. With this choice, the variance of the log travel times approaches δ as the trip distance increases, and equals $M + \delta$ for trips of length zero. The parameter λ controls how quickly the variability decreases towards δ. For our ambulance application, the unknown parameters in the model are then $\theta \triangleq (c, u_1, \ldots, u_L, \mu_1, \ldots, \mu_K, M, \delta, \lambda)$.

3.2.2 Estimation

We use a Bayesian formulation to estimate the parameters of the WT model. This allows uncertainty in the parameter estimates to be taken into account for travel time predictions. Predictions are based on the posterior distribution of the parameters, which is proportional to the prior density (specified below) times the likelihood function. The likelihood function is equal to the product over trips i of the lognormal density of T_i as specified in Equation 3.1. We estimate each parameter and relevant function of the parameters by its posterior mean, and summarize our uncertainty with a 95% interval estimate, the endpoints of which are the 0.025 and 0.975 quantiles of the posterior distribution. Computation of the posterior distribution is done via Markov chain Monte Carlo [56].

For our ambulance application, results are robust to moderate changes in the prior distributions for the unknown parameters $(c, u_1, \ldots, u_L, \mu_1, \ldots, \mu_K, M, \delta, \lambda)$, due to the large volume of data. Results are reported for the following prior dis-

tributions, with mutually independent parameters:

$$u_\ell \sim \mathcal{LN}(\nu_\ell, \xi_u^2), \quad \mu_k \sim \mathcal{N}(0, \xi_\mu^2), \quad \ell \in \{1, \ldots, L\}, \quad k \in \{1, \ldots, K\}$$
$$c \sim \text{Unif}(0, \infty), \quad \sqrt{M} \sim \text{Unif}(0, \infty), \quad \sqrt{\delta} \sim \text{Unif}(0, \infty), \quad \lambda \sim \text{Unif}(0, \infty).$$

The constant ν_ℓ is a prior estimate of the unit travel time on the log scale, for road class ℓ. For example, there might be initial speed estimates for each link in class ℓ, or perhaps known speed limits or recorded GPS speed data. In such cases, ν_ℓ can be set equal to the mean of the log inverse speeds. For the ambulance study, we use GPS speed data to specify a common ν_ℓ for all ℓ. The constant ξ_u captures how strongly we believe our prior estimate ν_ℓ of the log unit travel time. We take ξ_u to be large, allowing the information in the data to dominate the posterior estimate of u_ℓ. Specifically, we set ξ_u so that there is roughly 95% prior probability that u_ℓ is within a factor of two of e^{ν_ℓ}, which corresponds to $\xi_u = (\log 2)/2$. Similarly, ξ_μ captures our prior uncertainty in the value of μ_k, and by the same argument we set $\xi_\mu = (\log 2)/2$. We have no prior information about c, M, and δ, so we use uniform priors. Although these uniform prior distributions are non-integrable, the posterior distribution is integrable and valid. The uniform priors are on the square root of δ and M, because the square roots of these parameters are on the scale of the standard deviation of the log travel times, and it is more appropriate to put a uniform prior on a standard deviation than on a variance [15].

To estimate the posterior distribution for each parameter, we use a Metropolis-within-Gibbs Markov chain Monte Carlo method [56]. Specifically, we use Metropolis-Hastings (M-H) to update each of the unknown parameters in turn, conditional on the current values of the other unknown parameters. For example, to update the parameter u_ℓ, we propose a new value

$u_\ell^* \sim \mathcal{LN}(\log(u_\ell), \psi^2)$. The proposed sample is accepted with the appropriate M-H acceptance probability, which is the minimum of 1 and the following product of the prior, likelihood, and proposal density ratios:

$$\frac{\mathcal{LN}(u_\ell^*; \nu_\ell, \xi_u^2)}{\mathcal{LN}(u_\ell; \nu_\ell, \xi_u^2)} \frac{\mathcal{LN}(u_\ell; \log(u_\ell^*), \psi^2)}{\mathcal{LN}(u_\ell^*; \log(u_\ell), \psi^2)} \times \frac{\prod_{i=1}^{I} \mathcal{LN}\left(T_i; \mu_{k(i)} + \log\left(c + \sum_{j \in A_i} f_{ij} d_j u_{\ell(j)}^*\right), M e^{-\lambda d_i} + \delta\right)}{\prod_{i=1}^{I} \mathcal{LN}\left(T_i; \mu_{k(i)} + \log\left(c + \sum_{j \in A_i} f_{ij} d_j u_{\ell(j)}\right), M e^{-\lambda d_i} + \delta\right)}.$$

The variance ψ^2 is a constant that may be tuned to control the average acceptance probability, which theoretical evidence suggests should be roughly 23% for optimal efficiency [52]. Similarly, we use a lognormal M-H proposal to sample the parameter c. To sample the parameters μ_k ($k \neq 0$), M, λ, and δ, we use a normal distribution for the proposal.

To obtain the results in this chapter, we ran each Markov chain for 120,000 iterations, including a burn-in period of 20,000 iterations. To assess the Monte Carlo error, we calculated Monte Carlo standard errors for each of the parameter estimates, using batch means [30]. Standard errors are quite low, roughly 1-2% of the parameter estimate for the μ_k parameters and 0.03-0.2% for the other parameters. The computation time for each Markov chain iteration scales linearly with the number of vehicle trips, for a fixed road network. Each Markov chain run for these experiments takes roughly 18 hours on a personal computer, without utilizing parallel computing. Since the likelihood is a product over the terms for each trip, computation time could be decreased by calculating the likelihood terms in parallel batches. The Budge et al. nonparametric method [8] is estimated using maximum (penalized) likelihood [50] and is faster than our Bayesian implementation. In practice, however, ambulance travel time estimates are updated infrequently, so increased computation time is not a severe drawback [62].

3.3 Toronto EMS Data

We use our WT method to study ambulance travel times in Toronto, Ontario. The goal is to estimate the distribution of time required for an ambulance to drive to the scene of a high-priority emergency, in which case the ambulance travels at high "lights-and-sirens" speed. The data are provided by Toronto EMS (Emergency Medical Services), and include all such ambulance trips in Toronto during the years 2007 and 2008. We analyzed a subset of these data from the Leaside region of Toronto in the previous chapter [62]; here we estimate travel times on the entire Toronto road network, which consists of 68,272 links.

The data associated with each trip include the approximate start and end times and locations of the trip, as well as sparse GPS location and speed readings during the trip. The GPS measurements are stored every 200 meters (m) of travel or 240 seconds (s), whichever comes first (typically the distance constraint is satisfied first for lights-and-sirens travel).

Preprocessing the data is a substantial challenge, due to factors such as human error in recording the start and end times and locations of the trips, the presence of trips where the ambulance doubled back on itself, and the presence of GPS measurement error. These challenges and our preprocessing algorithm are described in Section 3.3.1. After preprocessing we are left with 157,283 ambulance trips, having removed 20,443 trips. The median shortest-path distance between the start and end locations is 2,530 m.

To apply our WT method, we first estimate the path traveled for each ambulance trip, using the sparse GPS data. Many such map-matching methods could be used [34, 36, 45, 46]; we use a variant of the one introduced in Chapter 4.

3.3.1 Preprocessing

For each ambulance trip, we have the time the ambulance departed for the emergency (the enroute time), the arrival time, and GPS readings recorded between those two times. Ideally, we would use the difference between the enroute and arrival times as the total trip travel time, and use the GPS readings to estimate the path traveled via a map-matching algorithm. However, the enroute and arrival times are error-prone. They are manually recorded inside the ambulance by a button push, and sometimes the button is pushed at the wrong time. For example, sometimes the button indicating arrival at the scene is not pushed until after the ambulance *departs* from the scene. The GPS device continues to record data, so there will be many consecutive readings with speed 0 in between the recorded enroute and arrival times, while the ambulance is parked at the scene. A stylized example of this issue is given in Figure 3.1.

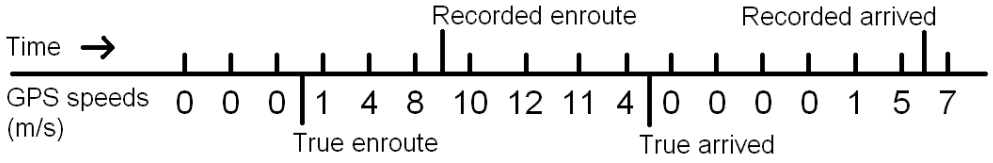

Figure 3.1: A stylized example of the effect of error in recorded enroute and arrived times.

Instead of using these error-prone enroute and arrival times, we estimate the start and end locations and times using the GPS data. First, to extract only the GPS readings where the ambulance was actually driving to the scene, we isolate the first "traveling block" (defined below) of GPS points, and discard the rest. Then we take the first and last GPS points of the traveling block as the estimated start and end locations and times of the trip. Due to GPS measurement error, these locations are not necessarily on the road network, but the map-matching

algorithm we use can handle this discrepancy [60].

A traveling block is a maximal consecutive sequence of GPS readings, with the requirements:

1. Begins and ends with a non-zero GPS speed.
2. Has at least 3 non-zero speed GPS readings.
3. Has no pair of GPS readings (consecutive or otherwise) with:
   (a) Timestamps at least 30 seconds apart but with average speed < 0.5 m/s, using straight-line distance.
   (b) Timestamps at least 2 minutes apart but with average speed < 2 m/s, using straight-line distance.
   (c) Average speed (straight-line) greater than 100 m/s.
4. Has straight-line distance of at least 400 m between the first and last GPS readings.
5. Has average speed (based on straight-line distance) between the first and last GPS readings no greater than 60 m/s.

Each of these requirements are designed to eliminate a certain type of error. Requirement 1 removes zero-speed GPS readings at the beginning or end of the trip. Requirement 2 ensures that we can estimate start and end locations for the trip, with at least one additional GPS reading for path estimation. Requirement 3 ensures that the trip does not have a long stationary period in the middle, as in Figure 3.1. This requirement also removes trips where the ambulance turned around, and subsequent GPS readings are very close to each other. While this is possible behavior, it is unhelpful for response time estimation to include these

trips. Finally, this requirement also removes trips with severe errors in the GPS timestamp or location. Occasionally the data contain successive GPS readings with identical timestamps but different locations, or GPS readings with impossible locations. Requirements 4 and 5 act similarly to Requirement 3, but on the entire trip. Requirement 4 removes trips where the ambulance turned around and the first and last GPS reading are very close to each other. Requirement 5 removes rare trips where the GPS locations are shifted by a very large amount from the true location.

3.3.2 Exploratory Analysis

Here we highlight exploratory analysis of the Toronto EMS data, after trip preprocessing. Results from this analysis motivate the modeling assumptions described in Section 3.2.1 for the ambulance study. After preprocessing, each trip consists of a sequence of GPS readings. To assist exploratory analysis of the travel time distribution between any two locations, we map the first and last GPS readings of each trip to the nearest intersections in the network, to use as estimated start/end locations (this differs from our travel time model, in which trips are allowed to start and end in the interior of links). We collect the most common pairs of start/end intersections for the trips in the dataset; there are 10 start/end pairs with at least 40 trips between them.

Figure 3.2 shows normal quantile-quantile (Q-Q) plots for the log travel times between the four most common start/end pairs in the dataset. The shortest-path distance (in meters) between the start and end locations is shown above each Q-Q plot. Also shown on the Q-Q plots are 95% pointwise confi-

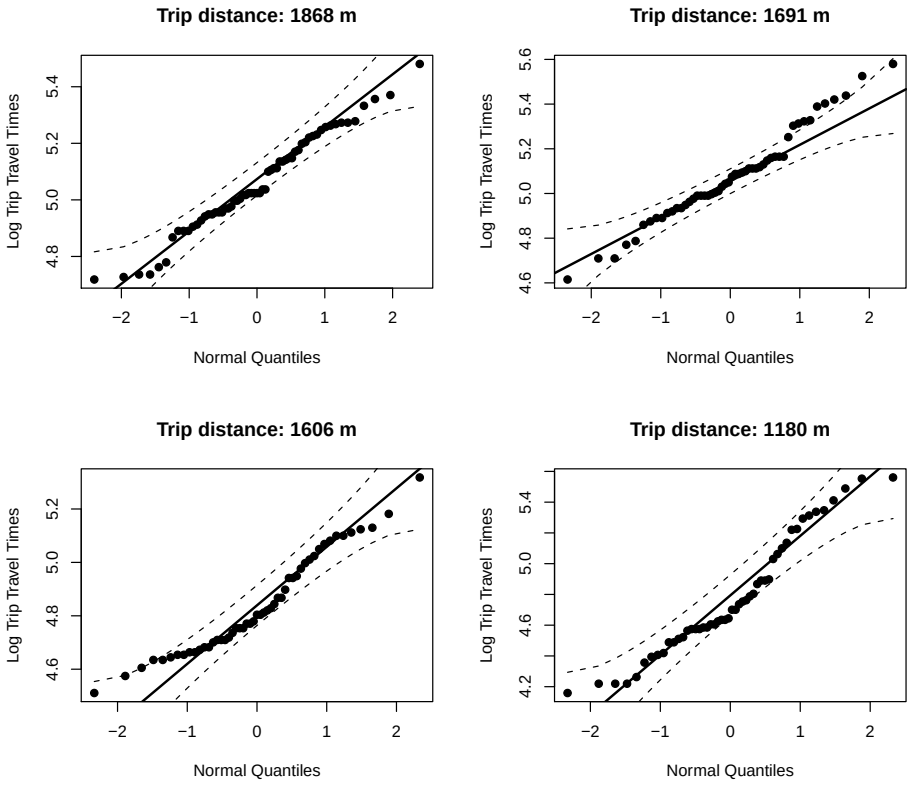

Figure 3.2: Normal quantile-quantile plots for travel times between the four most common start/end location pairs in the Toronto EMS dataset.

dence bands, under the null hypothesis that the log travel times are normally distributed. Only 6% of the observed travel times in the four plots fall outside the pointwise confidence bands, which suggests that the lognormal assumption is reasonable (if it is correct then we expect roughly 5% of the observations to fall outside of the bands). Although nearly all of these points occur on a single one of these four plots, this is not surprising because the points on a Q-Q plot are strongly dependent. Similar Q-Q plots can be constructed for the one-hundred most common start/end pairs, which range in shortest-path distance from 404 to 4,717 meters, and they also suggest lognormal travel times.

We also wish to investigate the variability in travel times for each start/end

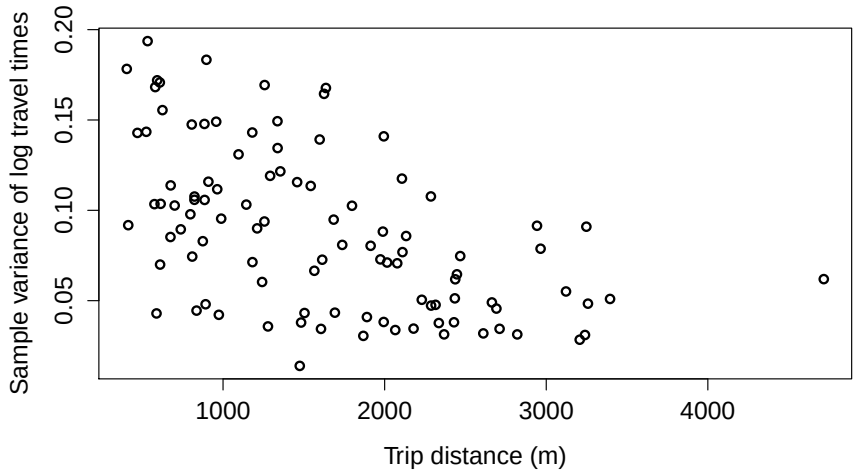

Figure 3.3: Sample variances of log travel times for the 100 most common start/end location pairs in the Toronto EMS dataset.

location pair. Figure 3.3 shows a scatterplot of the sample variance of the log trip travel times vs. the shortest-path distance, for the one-hundred most common start/end pairs. There is a general decreasing trend in the variance, the shape of which suggests the exponential decay model described in Section 3.2.1. This is for the log travel times; on the original scale, the variances increase with distance. We also construct a similar scatterplot where each point represents trips of a similar distance across the entire city, not just between specific locations. This plot is given in Figure 3.4. In this case, we again observe a decreasing trend, but with much less noise than in Figure 3.3. This is consistent with the results seen by Budge et al., who observed decreasing coefficient of variation of travel times with increasing distance. The line in Figure 3.4 is a fitted exponential decay according to the model proposed in Section 3.2.1. The fit is extremely good. The parameter estimates are $M = 0.22$, $\lambda = 0.0008$, and $\delta = 0.08$. We use these same estimated parameters in our map-matching method of Chapter 4.

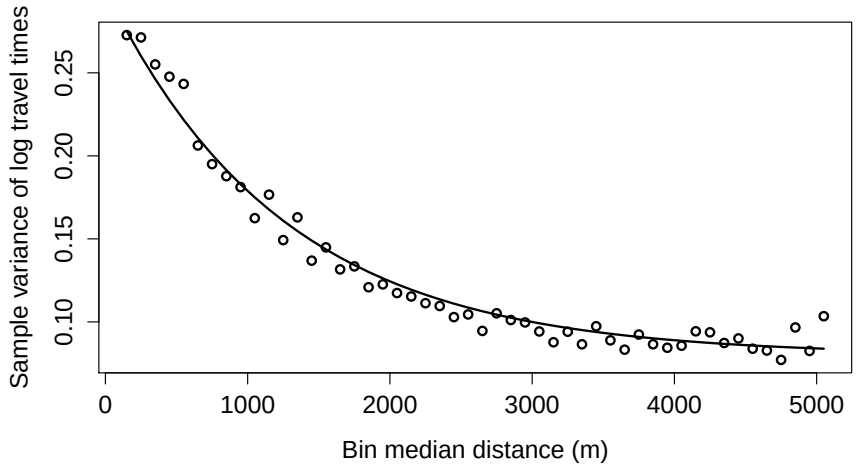

Figure 3.4: Sample variances of log travel times for Toronto ambulance trips, binned by shortest-path distance.

3.4 Application of TomTom

TomTom is a maker of navigation products. These products use both historical travel time averages and real-time traffic information from TomTom devices in vehicles to provide average travel time estimates between any two locations. These are intended for use by standard-speed vehicles, not ambulances traveling at lights-and-sirens speed. However, the TomTom estimates still provide a useful comparison. We report results using their historical average travel time estimates after adjusting for bias (see Section 3.5.1). Bias adjustment does not fully account for the differences between the TomTom context and ours; for example, intersection effects are much lower for L-S ambulances because they do not stop for red lights. Thus, our results should not be interpreted as an evaluation of the quality of TomTom's estimates. On the contrary, the fact that their estimates are competitive with the other methods (see Section 3.5.1) shows that standard vehicle data can be useful for predicting L-S travel times.

3.5 Results

Here we give the results of ambulance travel time estimation using the Toronto data. We compare our WT method, our IL method introduced in Chapter 2, the nonparametric method of Budge et al., and the TomTom predictions. For our WT method, we use seven road classes and four time bins. Class 1 corresponds to highways, Class 2 to major arterial roads, Classes 3-6 to smaller-sized roads in decreasing order, and Class 7 to highway on and off-ramps. Time Bin 0, the baseline bin, corresponds to weekday off-peak times (10 a.m. - 3 p.m., 7-10 p.m.), Bin 1 to rush hour (6-10 a.m., 3-7 p.m.), Bin 2 to weekend daytime (6 a.m. - 10 p.m.), and Bin 3 to late night (10 p.m. - 6 a.m.). We chose these bins by observing the change in average GPS speed readings across the week.

We split the ambulance trips randomly into two equal-sized sets, using half of the data to train the WT and Budge et al. methods, and the other half as test data for all the methods. Then we reverse the training and test halves. Results from these two experiments are similar.

u_1	u_2	u_3	u_4
0.0353 $[0.0343, 0.0363]$	0.0603 $[0.0600, 0.0606]$	0.0653 $[0.0648, 0.0659]$	0.0779 $[0.0769, 0.0791]$
u_5	u_6	u_7	μ_1
0.1018 $[0.0997, 0.1038]$	0.0712 $[0.0646, 0.0781]$	0.0450 $[0.0426, 0.0476]$	0.0268 $[0.0215, 0.0323]$
μ_2	μ_3	c	M
-0.0083 $[-0.0139, -0.0026]$	-0.0097 $[-0.0150, -0.0044]$	25.08 $[24.52, 25.66]$	0.2064 $[0.1932, 0.2203]$
δ	λ		
0.0576 $[0.0562, 0.0589]$	0.00097 $[0.00091, 0.00104]$		

Table 3.1: Parameter estimates from our WT model, along with 95% intervals expressing parameter uncertainty.

First we analyze parameter estimates from the WT method for the first training set, shown in Table 3.1. The road class parameter estimates appear reasonable. The estimated unit travel time $u_1 = 0.0353$ s/m for Class 1 (highways) corresponds to 102 km/hr. For Class 7 (highway on/off ramps), $u_7 = 80$ km/hr. For Class 2 (major arterial roads), $u_2 = 60$ km/hr. The estimated speeds decrease for smaller roads, except for Class 6, the smallest roads. These roads are relatively uncommon, and the interval estimate is wider for u_6 than for the other parameters, reflecting larger uncertainty in the value of u_6.

The rush hour time bin parameter estimate $\mu_1 = 0.0268$ corresponds to 2.7% larger travel times for rush hour, relative to the weekday off-peak bin. The estimates of μ_2 and μ_3 correspond to roughly 1% smaller travel times for weekend and late night, relative to weekday off-peak. All these values are close to zero, indicating that lights-and-sirens ambulance speeds are remarkably consistent across time bins, in contrast to standard travel speeds (see Section 3.5.3).

Our lognormal model implies that about 95% of trips are predicted to fall within two standard deviations of the median on the log scale, i.e. within factors of $e^{-2\times\text{SD}}$ and $e^{2\times\text{SD}}$ of the median on the original scale. Thus the variance estimate $\delta = 0.0576$ implies that for very long trips, about 95% of the travel times will be within factors of 0.62 and 1.6 of their median travel time. The estimate $M = 0.2064$ implies that for very short trips, about 95% of the travel times will be within factors of 0.36 and 2.8 of their median travel time.

3.5.1 Travel Time Prediction Comparison

Next we compare the predictive performance for our WT method, several reduced versions of the WT method, the nonparametric method of Budge et al. [8], and the TomTom estimates. Recalling that we use half of the data for training and the other half for testing and then reverse, here we evaluate the accuracy of the predicted travel time distribution for trips in the test data. For each test trip we evaluate the quality of a point estimate of the travel time, the predictive interval estimate, and the distribution estimate using appropriate statistical measures. For TomTom we only evaluate the quality of the point estimate, since interval and distribution estimates are not available. For our WT method and that of Budge et al., we use the median travel time as the point estimate of travel time. The 95% predictive interval from those methods is taken to be the estimated 0.025 and 0.975 quantiles of the travel time distribution.

When using the WT method to predict the travel time for the trips in the test data, we obtain predictions under two scenarios: (1) using the estimated route taken by the vehicle (based on the GPS data), or (2) not using this information. Using the estimated route emulates a situation in which we know the route that the driver will take, for instance if the driver were required to take a route specified by the dispatcher. Such control over the route could be desirable since then the route could be optimally selected using the most recent traffic conditions. However, most ambulance organizations leave the route choice to the driver. To emulate this situation, in Scenario 2 we predict the travel time without using the route information (only using the start and end locations of the trip). In this scenario we obtain an estimated fastest route according to the WT model (as described in Section 3.5.6), and base our predictions on this route.

Budge et al. base their travel time predictions on the shortest-path distance between the start and end locations [8]. In the spirit of Scenario 1, since we have estimated routes for each ambulance trip, it is natural to extend their method to use the distance of the estimated route, instead of the shortest-path distance. Therefore, we obtain predictions from their original method where the training and test sets both use the shortest-path distance, and the extended method where the training and test sets both use the estimated route.

We perform bias correction for each estimation method, since bias may be present for a variety of reasons. For example, bias arises in Scenario 2 because in this scenario our WT method treats the ambulance paths differently in the training and test data. For the training trips the estimated route is used, while for the test trips the fastest route is used, resulting in a tendency to underestimate travel times. Bias may also be present in each method due to inaccuracies of the assumed model. The TomTom estimates are severely biased, because they are intended for vehicles traveling at standard speeds, not lights-and-sirens speeds. We do bias correction on the log scale via cross-validation as described in the previous chapter (Section 2.5). Bias correction is done on the log scale to lessen the impact of outlying travel times.

Results are shown in Table 3.2. We report point estimation performance using the root mean squared error (RMSE, in seconds) of the point estimate compared to the true time, and using the RMSE of the log predictions compared to the true log time ("RMSE log"). Due to the inherent variability in travel times, even a perfect distribution estimate would have RMSE and RMSE log considerably above zero. We report the RMSE log because it is less affected by outlying travel times than the RMSE. Outliers are present for at least two reasons; first,

a small number of trips were apparently not driven at typical lights-and-sirens speeds, although they were recorded as high-priority trips. Second, some trips have high error in the recorded GPS locations, in which case the estimated path may be inaccurate.

Estimation method	RMSE (s)	RMSE log	Cov. %	Width (s)	CRPS (s)
WT, estimated route	72.3	0.298	94.4	218.9	34.6
WT, fastest route	77.7	0.322	93.1	219.7	37.3
WT, 1 var. param.	72.5	0.297	94.1	225.9	35.2
WT, 1 time bin	72.4	0.298	94.4	219.1	34.7
WT, 1 road class	76.8	0.312	94.3	231.0	36.7
Extended Budge et al.	74.9	0.302	94.6	229.1	35.7
Budge et al.	79.7	0.325	94.8	248.1	38.3
TomTom	82.1	0.347	NA	NA	NA

Table 3.2: Travel time prediction performance for the Toronto EMS lights-and-sirens data.

To evaluate the interval estimates, Table 3.2 shows the percentage of test trips where the observed travel time falls in the 95% predictive interval (the coverage, "Cov. %"), and the geometric mean width of the 95% predictive intervals ("Width"). Coverage close to or above 95% combined with small interval width is desirable, since it indicates that the predictive distribution is narrowly concentrated around the true travel time, while reflecting the true variability in travel times.

Table 3.2 evaluates the quality of the distribution estimates by reporting the continuous ranked probability score (CRPS) [18]. This is a "strictly proper" measure of distribution estimation accuracy, meaning that only a perfect distribution estimate achieves the lowest expected score [19]. If F is the estimated distribution function and x is the observed travel time, $\text{CRPS}(F; x) \triangleq \int_{-\infty}^{\infty} [F(y) - \mathbf{1}(y \geq x)]^2 \, dy$, i.e. the integrated square of the difference between F and the empirical distribution function based on the single observation x [18].

We report the mean CRPS over the test trips [18]; a lower value corresponds to better distribution estimates. Even a perfect distribution estimate would yield a CRPS well above zero, due to the inherent variability of travel times.

In Table 3.2, in addition to reporting the accuracy of our WT method under Scenarios 1 and 2, and the accuracy of the competing methods, we report the accuracy of several simplified versions of the WT method under Scenario 1. This indicates whether the reduced models are as effective as our full WT model and which aspects of our full model are the most important. We consider the following reduced models: (a) only one time bin, (b) only one road class, and (c) only one variability parameter instead of the exponential model.

As seen in Table 3.2, our WT method under Scenario 1 (using the estimated route) outperforms the Budge et al. method by 8-10% in RMSE, RMSE log, and CRPS, and outperforms the extended Budge et al. method by 1.5-3.5% in the same metrics. The WT method's interval estimates have almost identical coverage to those of Budge et al. but are narrower on average, by 12% compared to the original Budge et al. method and by 4.5% compared to the extended method. Under Scenario 2, the WT method outperforms the original Budge et al. method by 2.6% in CRPS and 1-3% in RMSE and RMSE log. The mean predictive interval width from the WT method under this scenario is 11% narrower than that of Budge et al., though with slightly lower coverage. These performance differences are most likely due to our model's inclusion of different speeds for the different road classes, as well as time effects.

The WT method outperforms the TomTom estimates by 12-14% in RMSE and RMSE log under Scenario 1, and by 5-7% in the same metrics under Scenario 2. Scenario 2 is the more natural comparison, because we do not spec-

ify the route traveled when obtaining the TomTom estimates, instead allowing TomTom to pick the optimal route. TomTom's estimates perform respectably, indicating that after bias correction, standard vehicle data do have predictive power for lights-and-sirens ambulance trips. In a similar experiment (not reported), we compared the WT method trained on lights-and-sirens data with the WT method trained on standard speed data, for predicting lights-and-sirens travel times, on the subregion of Toronto used in Chapter 2. We found a similar difference in performance as with the TomTom estimates; the model trained on lights-and-sirens data outperformed the model trained on standard data, which is not surprising since the test data were lights-and-sirens trips, but the performance of the standard data was respectable.

Regarding the reduced versions of the WT model, the method with only one time bin performs essentially as well in all metrics as the full method. We explore this observation in more detail in Section 3.5.3. The method with only one variability parameter performs as well in point estimation but slightly worse in distribution estimation than the full model. The method with only one road class performs worse than the full method and the other reduced methods in all metrics. It appears to be quite important to allow for varying speeds across road classes. The extended Budge et al. method outperforms our method with one road class. Both models rely only on travel distance; however, the Budge et al. method is more flexible than our WT method with one road class, because the point estimates on the log scale are not restricted to a linear function of distance.

3.5.2 Comparison to the IL Method

We also compare to our earlier IL method introduced in Chapter 2. The IL method is more computationally intensive than the WT method because it simultaneously estimates the paths traversed and realized link travel times for the historical trips, as well as the travel time parameters for each link. Because of this, we cannot apply it to the entire Toronto road network, so we compare our WT method to the IL method on the subregion of Leaside, Toronto, used to assess the IL method in Chapter 2. To ensure a fair comparison with previous results, we do not use the route information for the test trips (i.e., we use Scenario 2 from Section 3.5.1).

For application to the subregion, we make one minor change to the WT model introduced in Section 3.2.1. For the prior distribution on the variance parameter M, we use an exponential distribution with rate 5, instead of a uniform distribution. Since the dataset has few extremely short trips, posterior estimates of M are unstable unless we use a prior distribution that prefers smaller values. Failure to do this can lead to unrealistic travel time predictions for the few extremely short trips in the dataset.

Results are summarized in Table 3.3. We use the same five resamplings of training and test sets from the Toronto subregion data as in Chapter 2 (Section 2.7.3). The two methods perform roughly the same in terms of RMSE log, and the IL method performs only slightly better than the WT method in RMSE, even though the WT method is less computationally intensive. The WT method also has much better coverage of interval estimates than our IL method. This is because our IL method assumes independence between the travel times on different network links, which is unrealistic, as discussed in Section 3.1. Failing

to take into account the dependence between link travel times leads to underestimation of the variability in the total route travel time and thus overly narrow interval estimates.

Estimation method	RMSE (s)	RMSE log	Cov. %	Width (s)
IL, using fastest route	37.8	0.332	85.8	75.0
WT, using fastest route	38.1	0.331	91.3	90.3

Table 3.3: Travel time prediction performance of our WT method and IL method on the subregion of Leaside, Toronto.

3.5.3 Inflation of Time Effects

In Section 3.5.1, we observed that the inclusion of time effects did not noticeably improve performance of our WT method on the Toronto EMS data. For ambulance fleets in other municipalities and for non-ambulance contexts, the differences in travel times across time bins may be greater. For instance, although the difference in travel speeds between rush hour and non-rush hour on the full Toronto dataset is only about 4% (obtained by comparing GPS speed readings), this difference is 8% if one restricts to the Leaside subregion of Toronto, and is 16% for standard speed ambulance data on the Leaside subregion. In this section, we artificially inflate the travel times for trips in the rush hour time bin, to see what effect the inclusion of time bin factors has on performance if the differences across time bins are larger.

We multiply each trip travel time in the rush hour time bin by an artificial inflation factor and apply our WT method using both one time bin and four time bins. The inflation percentages used are 5% (inflation factor 1.05), 10%, and 20%. The estimated routes from the GPS data are used for the test trips (Scenario 1 in

Section 3.5.1). Results are given in Table 3.4. For small rush hour inflation, the difference between the 4 time bin model and the 1 time bin model is minimal. However, the difference increases in a nonlinear manner with the increasing inflation, and becomes fairly large (6%) at 20% inflation. We expect that our WT method with multiple time bins would show substantial improvement over the method with one time bin on a dataset where the travel time difference between rush hour and non-rush hour is 20% or more.

	Rush hour inflation percentage			
	No inflation	5%	10%	20%
WT method, 4 time bins	72.3	73.2	74.2	76.3
WT method, 1 time bin	72.4	73.9	75.9	81.0

Table 3.4: Travel time prediction performance (RMSE), with rush hour travel time inflation.

3.5.4 Closest Ambulance Post Comparison

In this section and the next, we consider the effect of using different travel time distribution estimates on ambulance fleet management. We assume a set of locations of available ambulances, and calculate which ambulance is estimated to be closest in terms of median travel time to each intersection in the city, according to our WT method and the Budge et al. method. If the two methods estimate different ambulances to be closest to a particular intersection, this would lead an ambulance dispatcher to assign different ambulances to respond to an emergency at that intersection, if the policy is to dispatch the closest ambulance [11].

We select a set of twenty-five representative ambulance post locations in Toronto, by examining the empirical distribution of start locations of ambulance trips (after data preprocessing), and choosing commonly-occurring locations.

These ambulance posts are chosen to illustrate and compare the travel time estimates from our method and the Budge et al. method, and are not indicative of actual ambulance coverage of Toronto EMS.

For our WT method, we define the closest post to an intersection to be the one with the smallest estimated median travel time. This corresponds to Scenario 2 from Section 3.5.1, since we do not know the route that the ambulance will take. For the Budge et al. method, we use the closest post in shortest-path distance. Typically this coincides with the closest post according to median travel time, since their method models median travel time as a function of only shortest-path distance. However, it is not guaranteed since Budge et al. do not restrict the function to be increasing in distance, and for the Toronto data the estimated function does have small non-monotonic fluctuations.

In Figure 3.5, black points mark the intersections that are estimated to be closest to different posts, according to our WT method and the Budge et al. method. Light gray points represent the remaining intersections. The ambulance post locations are shown as black X's. Roughly 5% of the intersections in the city are estimated to be closest to different posts. Typically, only intersections that are roughly halfway between two posts have a chance to be marked. Therefore, if the number of ambulance posts were higher, it is likely we would see even more intersections marked. By comparison, roughly 10% of the intersections in the city are closest to different posts according to straight-line distance and shortest-path distance. Therefore, the 5% we observe comparing the WT method and the Budge et al. method is fairly large.

Figure 3.5: Intersections (shown in black) where the closest ambulance post differs when estimated by our WT method and the Budge et al. method, with post locations shown as X's.

3.5.5 Probability of Arrival Within a Time Threshold

In this section, we calculate the probability that an ambulance is able to reach each intersection in the city within a time threshold, given a set of currently available ambulance locations and a travel time distribution estimate for any path. Visual displays of these probabilities are called probability-of-coverage maps, and are useful to EMS practitioners [8]. We use the same set of twenty-five representative ambulance posts and the same methods for estimating the closest post to each intersection as in the previous section.

In Figure 3.6, we plot the probability that an ambulance arrives at each inter-

Figure 3.6: Probability of arriving at each intersection in Toronto from the closest ambulance post within 4 minutes, estimated by our WT method.

section in Toronto from the closest ambulance post within 4 minutes, according to our WT method. Each intersection is shaded in gray according to this probability, where darker points correspond to higher probability. The post locations are shown as white X's. The probability of arrival is very high for intersections near the closest post and becomes lower for intersections farther away.

The arrival probabilities from our WT method do not decrease solely as a function of travel distance from the closest post, but also incorporate road speeds. This becomes clear in the top panel of Figure 3.7, where we plot the differences between the arrival probabilities for our method and the Budge et al. method. The black points represent intersections where our method gives

at least 15% higher probability of arrival within 4 minutes than the Budge et al. method does. Thus, there is a substantial predictive difference between the two distributions for these intersections. The medium grey points represent intersections where the Budge et al. method gives at least 15% higher probability than our method does. The light gray points represent all other intersections. The ambulance post locations are again shown as black X's.

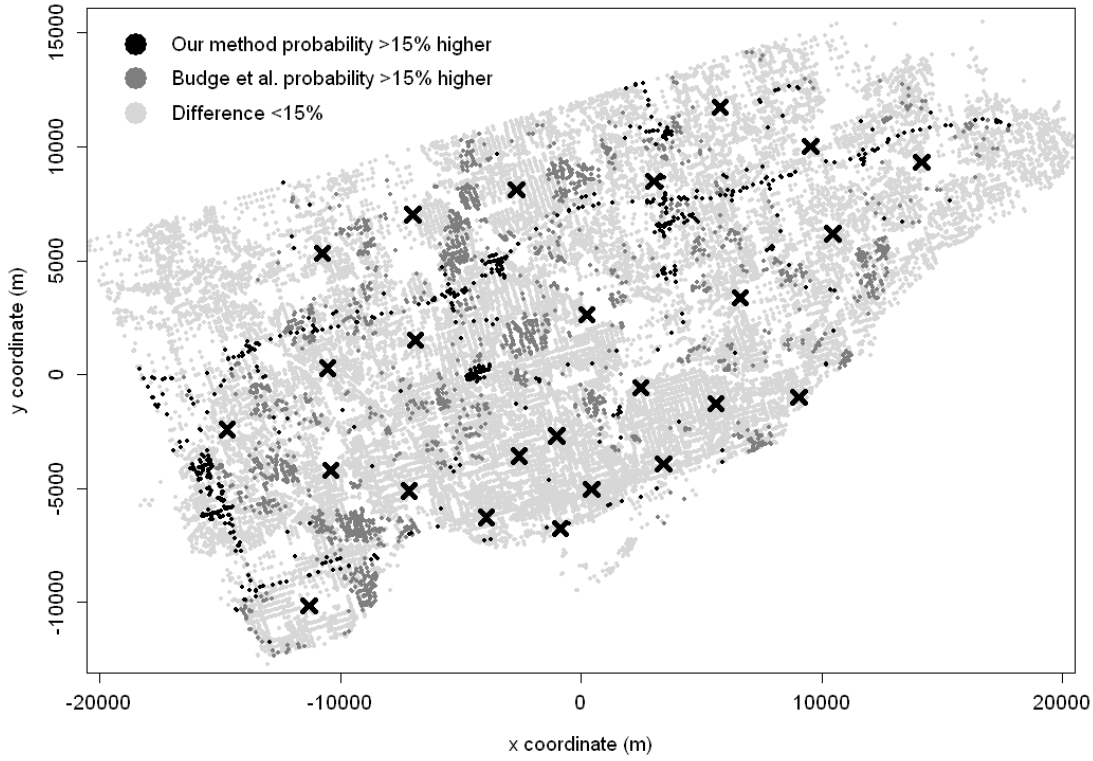

Figure 3.7: Differences in the estimated probability of arriving within 4 minutes, between our WT method and that of Budge et al.

Most of the intersections that are close to an ambulance post do not differ by 15% or more according to the two methods, because arrival probabilities from both methods are high. Similarly, intersections that are far from all ambulance posts also differ by less than 15%. On the other hand, many of the intersections that are at an intermediate distance to the closest ambulance post differ in ar-

rival probability by 15% or more. In fact, this is true for roughly 10% of all the intersections in the city. Many of the points where the probability from the WT method is at least 15% higher are on or near highways, particularly Highway 401, which is visible in Figure 3.7 as a sequence of black points running horizontally across the middle of the city. The highway road class speed estimate is high, so the method predicts better coverage when a highway can be used. There is another large collection of black points at the left edge of the figure that are close to Highway 427.

Many of the intersections where the Budge et al. probability is at least 15% higher are in residential areas where there is no direct path following highways or major arterial roads. For example, there are no major roads traveling from an ambulance post to the collection of gray points near location (-10000, -7000). Similarly, there is no direct route from an ambulance post to the collection of gray points near location (-5000, 7000). Though there are major arterial roads in the area, it would require a detour to use one. There are smaller roads that take more direct routes, but they have slower speed estimates.

3.5.6 Fastest Path Estimation

Here we describe the fastest path estimation for our WT method under Scenario 2 of Section 3.5.1. As noted in Section 3.3.1, the recorded start and end times for the ambulance trips are error-prone, so the first and last GPS readings in the first traveling block of the trip are used for the start and end times and locations. Since these two locations are not necessarily on the road network, to estimate the fastest path we first find the two nearest links to these GPS loca-

tions, and use the nearest points on these links as possible start/end locations. These links typically correspond to the two travel directions of the nearest road. For each of the four start/end location pairs, we calculate the fastest path in median travel time. Of these four possible paths, we use the one with the smallest median travel time as the estimated path. This method ensures that we obtain a reasonable path for each trip, which can begin or end in the interior of a link, and is not hampered by choosing the "wrong direction" of the nearest link.

3.6 Conclusions

We introduced a parametric model for estimating the distribution of vehicle travel times between any locations in a city. This method, called the WT method, is computationally tractable for large road networks and large datasets of vehicle trips, and is particularly useful when travel time data for individual roads in the city are sparse. The model parameters are interpretable, and include effects for the roads traveled by the vehicle and trip-level effects such as time of day. We used a Bayesian formulation and Markov chain Monte Carlo method to estimate the model parameters.

We tested the method on a large dataset of ambulance trips from Toronto. Exploratory analysis of the data indicated that the distribution of ambulance travel times between two fixed locations is well modeled by a lognormal distribution, with variability parameter depending on travel distance. These observations influenced our modeling choices. We compared travel time predictions from the WT method with predictions from a method published by Budge et al. [8] and commercially-available estimates from TomTom. We found that the WT method

outperformed the alternative methods in both point estimation and distribution estimation. We also compared the WT method with the IL method from Chapter 2 on a subregion of Toronto, and found that the WT method performed almost as well in point estimation and better in interval estimation.

We also investigated several reduced versions the WT method, to determine which features were the most important. The largest benefit came from the inclusion of parameters for each road class in the city, compared to a model with only one road class. However, there was little benefit in performance from adding multiple time bins across the week vs. a single time bin. In the Toronto dataset, the ambulance travel times do not vary substantially across the day and week, even during rush hour. Because other cities or datasets may be more variable in time, we performed an additional set of experiments by artificially inflating the difference in travel times between time bins. We found that if the travel times during rush hour were increased by at least 20%, then time bin factors provided a substantial benefit to estimation.

Finally, we investigated operational differences for ambulance fleet management from using the WT vs. Budge et al. methods. We fixed a set of representative ambulance posts in Toronto, and calculated the closest post to each intersection in the city, according to travel time estimates from each methods. We found that the two methods estimated 5% of the intersections in the city to be closest to different posts, which could lead to different dispatch decisions for emergencies at these intersections. We also calculated the probability that an ambulance arrives at each intersection in the city within 4 minutes, responding from the closest post. We found that for 10% of the intersections in the city, the two methods gave arrival probabilities that differed by more than 15%.

CHAPTER 4

A MONTE CARLO METHOD FOR MAP-MATCHING, WITH GPS BIAS ESTIMATION

4.1 Introduction

Map-matching refers to the problem of estimating the sequence of roads traveled by a vehicle from a set of locations and times recorded during the trip, for example by a Global Positioning System (GPS) device [63]. Map-matching is performed both on-line, where an estimated path is constructed as GPS readings are obtained [41], and off-line, where the path is estimated after the fact [67]. Map-matching is difficult particularly when the GPS data are sparse and error-prone. Error in GPS location observations can sometimes be very large [5, 9], on the order of 100 meters or more. Sparsity is often introduced to reduce data transmission and storage costs [41, 46]. Interest in map-matching techniques for sparse data is currently very high, because there is an explosion in the amount of this type of data available, from smartphones and GPS devices in taxis, ambulances, and other vehicles [5, 26].

There have been a large number of methods proposed for both on-line and off-line map-matching. Map-matching algorithms typically integrate geometric considerations, such as the distance of each GPS reading to the nearby links (road segments), with topological information, such as the length and characteristics of candidate paths, to create an overall rating for each candidate path, and choose the path with the highest rating [57]. Probabilistic models [5, 26] and Bayesian inference [44, 62] are also used. For general reviews and discussion, see Quddus, Ochieng, and Noland [45], and Wei et al. [59].

Recently, Bierlaire, Chen and Newman [5] introduced a map-matching method that returns a probability for each candidate path. Other recent off-line methods have been introduced by Lou et al. [34], who combine the geometric and topological methods introduced above with speed and time considerations and Rahmani and Koutsopoulos [46], who generalize and improve the method of Lou et al. We discuss these methods in more detail in Section 4.3.1. Recent on-line methods have been introduced by Miwa et al. [41], who use geometric and probabilistic considerations and investigate the possibility of using the empirical distribution of GPS location errors, and Hunter, Abbeel, and Bayen [26], who use a Conditional Random Field framework to integrate path selection models with probabilistic information about the GPS readings.

It has been observed that successive GPS location errors appear to be dependent, in the form of a persistent bias in a particular direction [31, 66]. Xu et al. observed that the GPS bias was fairly stable in the short term and changed smoothly on the time-scale of minutes [66]. There are several reasons why the locations have persistent bias. First, the digital road network is modeled as a collection of line segments with no width, which can cause up to several meters of apparent error. The road network may also contain errors that can lead to bias, such as roads that are missing or incorrectly defined as being one-way. Inherent GPS errors can also lead to bias, such as atmospheric delay [31] and the use of dead-reckoning in cases where GPS satellites cannot be observed [66].

Persistent GPS bias and random noise have been studied and corrected for via Kalman filters in the high-frequency GPS setting, notably by Kim, Jee, and Lee [31] and Xu et al. [66]. However, in probabilistic map-matching methods for sparse GPS data, the GPS errors are typically assumed to be independent and

normally distributed [5, 26, 34, 36, 62]. Hunter et al. [26] noted that it would be interesting to consider the exponential distribution as a more robust alternative.

In this chapter, we first investigate whether the path traveled and GPS location bias are identifiable, i.e. whether they can be uniquely determined given sufficient data. Assuming that there is no independent GPS error, we show that the path and GPS bias are identifiable up to translations of the path in the road network. However, even in cases where there is no alternative path in the road network that is a translation of the true path, it may not be possible to distinguish accurately the true path and bias from alternatives, given only a finite amount of GPS data.

Next, we investigate whether directly modeling the GPS bias can lead to improvements in map-matching performance for sparse data. Given a set of historical vehicle trips, we introduce a map-matching model where the GPS error consists of a bias vector, which is unchanging for all readings in a trip, plus an independent error for each reading. We treat the unknown path traveled and bias for each trip as missing data. We assume that the bias magnitude follows an exponential distribution with unknown mean, and that the independent GPS error for each reading also follows an exponential distribution with unknown mean. Thus there are two estimation problems to solve: (1) estimating the path and bias vectors for each historical trip, and (2) estimating the parameters of the bias and independent error distributions. We introduce a Bayesian model to estimate solutions to these two problems simultaneously. After estimating solutions to the two problems for the historical data, we can also estimate paths and biases for new vehicle trips by taking point estimates for the parameters of the bias and independent error distributions.

We use a Metropolis-within-Gibbs framework to estimate the missing data and unknown error parameters [56]. To draw samples of the path for each vehicle trip, we use the local Metropolis-Hastings (M-H) proposal introduced in Chapter 2. The problem of sampling paths on a road network via a Markov chain was recently addressed by Flötteröd and Bierlaire [14]. They form a M-H proposal by selecting a portion of the path to update, and re-routing that portion through a new node. We compare our method with theirs in Section 4.3.1.

Our Bayesian method gives posterior probabilities for each candidate path, as in Bierlaire, Chen and Newman [5] and our Chapter 2 [62]. This is particularly useful in applications that do not require a single path estimate, such as route choice models [5]. Unlike Bierlaire, Chen and Newman, we do not use a uniform prior distribution on the path traveled, but use the missing data model to capture the fact that faster paths are preferred. We model the path via a multinomial logit choice model on the expected travel time [39, 62].

We test our map-matching method on ambulance trips from Toronto and on simulated data on the same road network. We compare our method to a reduced method where there is no model of the GPS location bias, only independent GPS errors. We give evidence on the Toronto ambulance data that the GPS error has substantial persistent bias. We find that the full method with both GPS bias and independent error outperforms the reduced method in true and false positive rates on simulated data. We also discuss specific types of paths from the real and simulated data where the full model performs better.

This chapter is organized as follows. In Section 4.3, we introduce our map-matching model and estimation method. In Section 4.4, we perform experiments on the Toronto ambulance data, highlighting exploratory analysis in Sec-

tion 4.4.1 and assessing the results of map-matching in Section 4.4.2. In Section 4.5, we compare the full and reduced models in experiments on simulated data. We draw conclusions in Section 4.6.

4.2 GPS Bias and Error Identifiability

In this section, we consider whether the path traveled and GPS location bias are identifiable, i.e. whether they can uniquely be determined given sufficient GPS data. Assume that there is no independent GPS error, only unchanging bias for all readings in a trip. We show that the path and bias are identifiable up to translations of the path in the road network. If there are two paths in the road network that differ by a translation vector, then the path and bias are unidentifiable. However, this is the only circumstance that leads to unidentifiability.

To make this precise, we first give basic definitions of a link, road network, and path. A *link* is a piecewise linear curve in $\mathbb{R}^2$, made up of a finite number of closed, finite length line segments in $\mathbb{R}^2$, intersecting at their endpoints. This is the standard definition that is used in practice [46]. A *road network* is a finite collection of links and intersections between them, where links intersect only at their overall endpoints. A *path* is a closed, continuous, piecewise linear curve made up of a sequence of links in a road network. A path need not begin or end at intersections; the first and last links in the path may be used only fractionally.

Denote a path P translated by a vector b as $P + b = \{z + b | z \in P\}$. Let $\mu_{P,b}$ denote the uniform measure on the translated path $P+b$. Precisely, for a set $A \in \mathcal{B}$, where $\mathcal{B}$ is the Borel σ-algebra on $\mathbb{R}^2$, define $\mu_{P,b}(A) = \lambda(A \cap (P+b))/\lambda(P+b)$, where $\lambda(S)$ is the 1-dimensional Lebesgue measure of a curve S [49].

We have defined a map from a parameter space of paths and bias vectors to the uniform measures on piecewise linear curves in $\mathbb{R}^2$, i.e. a parametrization $g : (P, b) \to \mu_{P,b}$. We now consider whether this parametrization g is identifiable. A parametrization is unidentifiable if different values of the parameters map to the same probability measure [4], i.e. in our case if there are (P, b) and (P^*, b^*) with $(P, b) \neq (P^*, b^*)$ but $\mu_{P,b}(A) = \mu_{P^*,b^*}(A)$ for all $A \in \mathcal{B}$.

Lemma 4.2.1. Fix a road network. If there are two paths P and P^* such that $P = P^* + v$ for a vector $v \neq 0$, then the parametrization g is unidentifiable.

Proof. Fix any bias vector $b \in \mathbb{R}^2$ and consider the translated path $P + b$ and measure $\mu_{P,b}$. Since the translated path $P + b = P^* + b + v$, we must have $\mu_{P,b}(A) = \mu_{P^*,b+v}(A)$ for all $A \in \mathcal{B}$. Therefore since $(P, b) \neq (P^*, b + v)$, the parametrization g is unidentifiable. □

In practice, if paths can begin and end in the interior of links, as we allow, then the parametrization g will always be unidentifiable. To show this, take any link in the road network and any line segment S that is part of that link. Let P be a continuous portion of S, but not all of S, and let P^* be a different portion of S of the same length as P. Then $P = P^* + v$ for some $v \neq 0$, and so the parametrization g is unidentifiable by Lemma 4.2.1.

The lack of identifiability shown in Lemma 4.2.1 is restricted to paths that are translations of each other. We formalize this in Lemma 4.2.2.

Lemma 4.2.2. The parametrization g is identifiable up to translations between paths. That is, for paths and biases (P, b) and (P^*, b^*) such that $(P, b) \neq (P^*, b^*)$ and $\mu_{P,b}(A) = \mu_{P^*,b^*}(A)$ for all $A \in \mathcal{B}$, we must have $P + b = P^* + b^*$.

Proof. Take any $(P, b) \neq (P^*, b^*)$ such that $P + b \neq P^* + b^*$. Without loss of generality, assume that there exists $y \in \mathbb{R}^2$ such that $y \in P + b$ but $y \notin P^* + b^*$. Since paths are closed, there must be a neighborhood $D \in \mathcal{B}$ with $y \in D$ but $D \cap (P^* + b^*) = \emptyset$, and so $\mu_{P^*,b^*}(D) = 0$. However since paths are continuous, $\mu_{P,b}(D) > 0$. Thus, we cannot have $\mu_{P,b}(A) = \mu_{P^*,b^*}(A)$ for all $A \in \mathcal{B}$. □

For clarity, we consider examples in Figure 4.1. In the left panel, two possible paths P and P^* are shown by dotted lines, with corresponding bias vectors b and b^* shown as dashed lines. Example GPS readings from the distribution $\mu_{P,b}$ are shown as black dots, with corresponding locations on P and P^* as white dots. The path P^* is a translation of P, and so $\mu_{P,b}(A) = \mu_{P^*,b^*}(A)$ for all $A \in \mathcal{B}$. The path and bias cannot be distinguished between alternatives (P, b) and (P^*, b^*).

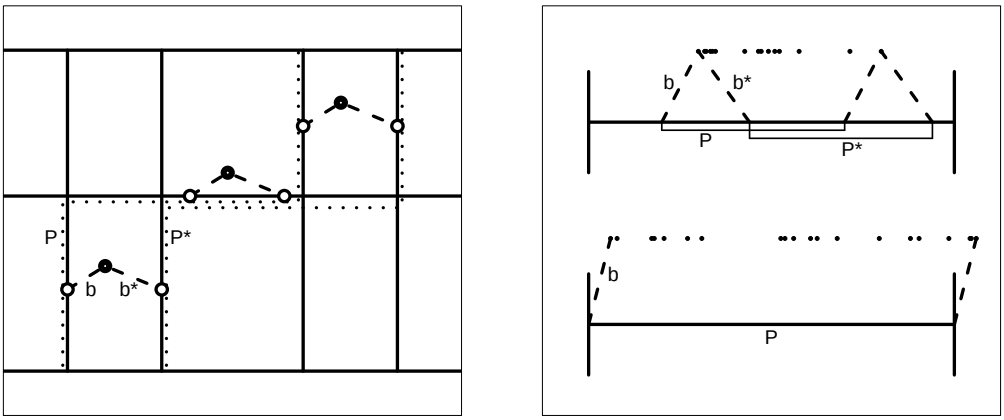

Figure 4.1: Stylized examples with GPS location bias, but not independent error.

Next, consider the top example in the right panel of Figure 4.1. The path P travels part of the way along the road and the path P^* travels the same distance but translated by a vector. The biases b and b^* are again shown as dashed lines. Since P^* is a translation of P, GPS locations sampled from $\mu_{P,b}$ (shown as small black dots) again cannot distinguish between (P, b) and (P^*, b^*).

Finally, consider the bottom path in the right panel of Figure 4.1. The path P now encompasses the entire road between the two intersections. Assume that there is no other road in the network traveling in the same direction that is at least as long as this road, so there is no path P^* that is a translation of P. In this case, (P, b) uniquely determines the distribution $\mu_{P,b}$.

There may be an alternative path and bias that cannot be distinguished from the true ones, given a finite number of GPS readings, even if there is no alternative path that is a translation of the true path. For example, again consider the bottom path in the right panel of Figure 4.1. For any finite sample of GPS readings from the uniform distribution $\mu_{P,b}$, there will not be readings with true locations exactly at either of the intersections. Therefore P will be indistinguishable from any other path along this road that is at least as long as the maximum distance between observed GPS locations. However, for any alternative path P^* on this road, if we sample repeatedly from $\mu_{P,b}$, with probability 1 we will eventually find two GPS readings that are separated by a distance larger than the length of P^*, and will conclude that P^* is impossible.

We believe the results in this section can be extended to the case where there is bivariate independent GPS location error as well as bias, given conditions on the independent error distribution. Formalizing this is a matter of current work.

4.3 Modeling and Estimation

In this section, we introduce a statistical model and estimation method for map-matching, given a set of historical vehicle trips. We treat the unknown path traveled and GPS bias for each trip as missing data. We use a Bayesian approach to

estimate the missing data and the parameters of the GPS bias and independent error distributions simultaneously. Thus, we obtain a posterior distribution on the path traveled for each trip in the historical dataset, the unchanging GPS bias for each trip, and the GPS bias and error distribution parameters.

To obtain a point estimate of the path driven for each trip, we use the maximum a-posteriori (MAP) estimate, i.e. the most common path in the posterior samples. The posterior distribution over paths gives us an understanding of the uncertainty in the path estimate for each trip. For example, we can compare the posterior probability of the MAP estimate with the posterior probability of the next most likely path. Sometimes the MAP estimate is far more likely than all other paths, but sometimes this ratio is close to 1. The marginal posterior probabilities for each link are also useful, as we saw in the examples of Sections 2.6.3 and 2.7.5, because they can highlight which portions of the path are uncertain.

4.3.1 Map-Matching Model

Here we introduce the statistical model used in our map-matching method. We use a model that assesses the same characteristics of potential map-matching solution paths as the models used by Lou et al. and Rahmani et al. [34, 46]. Lou et al. emphasized that a map-matching solution should use geometric, topological, and temporal considerations. Abstractly, we summarize these three concerns as follows. The estimated path should be:

1. Close to the GPS locations.
2. Short in length and reasonable for a driver to follow.

3. Similar in speed to average speeds for the roads used.

In Lou et al., these characteristics were captured by:

1. A normal distribution on the distance between each GPS location and its candidate point (a possible true location on the road network), independently across GPS readings.
2. For each pair of consecutive GPS readings, the ratio of the Euclidean distance between candidate points and the length of the shortest-distance path between candidate points.
3. A cosine distance function to compare the average speed on the shortest-distance path between candidate points to a typical speed for that road.

These three components were combined by Lou et al. into a rating for each candidate path. Rahmani et al. [46] used a similar but more general model, allowing the rating for a candidate path to be an arbitrary function of the characteristics of the path, for example the overall length or expected travel time, the number of left or right turns, or the type of roads used.

Our statistical model on the path traveled and GPS observations assesses the same three characteristics. We use:

1. An exponential distribution for the magnitude of the unknown GPS bias, with a uniform random direction, and an exponential distribution on the remaining distance for each reading to the nearest location on the path.
2. A multinomial logit choice model for the unknown path traveled, as a function of the expected travel time. Paths with shorter expected travel times have higher probabilities (see Section 2.2.1).

3. A lognormal distribution for the travel time between successive GPS observations, with mean equal to the expected travel time on the roads between the GPS observations.

We now give the details of this model. Consider a road network with J links, where link j has length $d(j)$, and a set of I vehicle trips on this network to be map-matched. The only data for each trip are GPS observations. In particular, we do not know the start and end times or locations. The start and end locations can be anywhere on the road network, not necessarily at intersections. This corresponds to the setting of our Toronto data in Chapter 3. Path i has GPS data $G_i = \left\{Z_i^\ell, t_i^\ell\right\}_{\ell=1}^{m_i}$, where m_i is the number of GPS observations, $Z_i^\ell = \left(X_i^\ell, Y_i^\ell\right)$ is the location of reading ℓ, and t_i^ℓ is the timestamp. We do not assume any other GPS information. Speed and heading observations are also useful for map-matching, but they are not always available [41].

The unknown path traveled and GPS bias are treated as missing data. Denote the path traveled in trip i as $A_i = \left\{A_i^1, \ldots, A_i^{N_i}\right\}$, where N_i is the number of links traversed in the path. The path A_i follows a multinomial logit choice model, as described below. Denote the bias vector $B_i = \{R_i, \theta_i\}$, with magnitude R_i and direction θ_i. The bias magnitude R_i follows an exponential distribution, $R_i \sim \text{Exp}(1/\mu_B)$, parameterized by the mean μ_B, and the direction θ_i follows a uniform distribution, $\theta_i \sim \text{Unif}(0, 2\pi)$.

We assume that the true location of the vehicle at time t_i^ℓ is the closest point on the path to the bias-removed location $Z_i^\ell - B_i$, with the restriction that the GPS readings must occur in their observed sequence. The distance D_i^ℓ between the bias-removed location $Z_i^\ell - B_i$ and the closest point on the path follows an exponential distribution: $D_i^\ell \sim \text{Exp}\,(1/\mu_E)$, parameterized by the mean μ_E. The

assumption of the closest point on the path is discussed below.

Next, we denote between-GPS times $\Delta t_i^\ell = t_i^{\ell+1} - t_i^\ell$, for $\ell = 1, \ldots, m_i - 1$. We also define expected between-GPS times $\{e_i^\ell\}_{\ell=1}^{m_i-1}$, given the path A_i, in the following manner. First, we assume there is an expected travel time $\tau(j)$ for each link j in the network. We discuss how these can be obtained in Section 4.3.5. Suppose GPS readings ℓ and $\ell+1$ were generated from links $A_{i,p}$ and $A_{i,q}$. Then the expected travel time between the GPS readings is

$$e_i^\ell = (1 - f(A_i^p, \ell))\tau(A_i^p, \ell) + \left(\sum_{k=p+1}^{q-1} \tau(A_i^k) \right) + f(A_i^q, \ell+1)\tau(A_i^q) + c_{\mathrm{L}}^\ell p_{\mathrm{L}} + c_{\mathrm{R}}^\ell p_{\mathrm{R}},$$

where $f(A_i^p, \ell)$ and $f(A_i^q, \ell+1)$ are the fraction of the length of links $A_{i,p}$ and $A_{i,q}$ before the true locations for readings ℓ and $\ell+1$. The terms $c_{\mathrm{L}}^\ell p_{\mathrm{L}}$ and $c_{\mathrm{R}}^\ell p_{\mathrm{R}}$ are turn penalties, where c_{L}^ℓ is the number of left (resp. right) turns between readings ℓ and $\ell+1$, and p_{L} is the penalty (in seconds) for a left (resp. right) turn. The penalties p_{L} and p_{R} are discussed in Section 4.3.5.

The between-GPS travel time Δt_i^ℓ follows a lognormal distribution with mean e_i^ℓ. Specifically, $\Delta t_i^\ell \sim \mathcal{LN}\left(\log(e_i^\ell) - \sigma_{i\ell}^2/2, \sigma_{i\ell}^2\right)$. The variance parameter $\sigma_{i\ell}^2$ is a function of the distance traveled between readings ℓ and $\ell+1$, which is denoted $\Delta d_i^\ell = (1 - f(A_i^p, \ell))\, d(A_i^p) + \left(\sum_{k=p+1}^{q-1} d(A_i^k)\right) + f\left(A_i^q, \ell+1\right) d(A_i^q)$. We describe how $\sigma_{i\ell}^2$ is estimated in Section 4.3.5.

For the model of the missing path data A_i, we use a multinomial logit choice model on the expected travel time [39]. This gives probability

$$\pi(A_i) = \frac{\exp\left\{-C\left((1 - f(A_i^1, 1))\tau(A_i^1) + \left(\sum_{k=2}^{N_i-1} \tau(A_i^k)\right) + f\left(A_i^{N_i}, m_i\right)\tau(A_i^1)\right)\right\}}{\sum_{a_i \in \mathcal{P}_i} \exp\left\{-C\left((1 - f(a_i^1, 1))\tau(a_i^1) + \left(\sum_{k=2}^{n_i-1} \tau(a_i^k)\right) + f(a_i^{n_i}, m_i)\tau(a_i^1)\right)\right\}},$$

where as above $f(A_i^k, \ell)$ denotes the fraction of link A_i^k before the true location of GPS reading ℓ, the set $\mathcal{P}_i$ contains all possible paths between the start and

end location of trip i, and $C > 0$ is a positive constant. There is a subtlety with $\mathcal{P}_i$, because the start and end links are not assumed known and can change. We resolve this issue by taking a set of possible actual start and end links. We do not have to evaluate the denominator of $\pi(A_i)$, because we only calculate the ratio of priors for two alternative paths.

This results in the following complete-data likelihood for the missing data $\{A_i, B_i\}_{i=1}^I$ and GPS data $\{G_i\}_{i=1}^I$, given the GPS error parameters μ_B and μ_E:

$$\mathcal{L}\left(\{A_i, B_i, G_i\}_{i=1}^I \middle| \mu_B, \mu_E\right) = \prod_{i=1}^{I} \Bigg[\pi(A_i) \text{Exp}\left(R_i; 1/\mu_B\right) \prod_{\ell=1}^{m_i} \text{Exp}\left(D_i^\ell; 1/\mu_E\right) \times \prod_{\ell=1}^{m_i-1} \mathcal{LN}\left(\Delta t_i^\ell; \log(e_i^\ell) - \sigma_{i\ell}^2/2, \sigma_{i\ell}^2\right) \Bigg]. \quad (4.1)$$

To complete the Bayesian model, we require prior distributions on the GPS error parameters μ_B and μ_E. We use improper uniform priors. We find that the choice of prior for these parameters, whether improper or proper (for example, an exponential distribution), makes little difference in estimation.

The other constants and parameters in the model are not estimated in a Bayesian manner, but are assumed fixed and known. These include $\sigma_{i,\ell}^2$ for each trip i and GPS reading $\ell \in \{1, \ldots, m_i - 1\}$, the turn penalties p_{L} and p_{R}, and the multinomial logit choice model constant C. We discuss how these parameters are set in Section 4.3.5.

Finally, we discuss the assumption that the GPS reading was generated at the closest point on the path to the bias-removed GPS location. Without this assumption, it is necessary to estimate where the vehicle is at all times, in order to calculate the GPS location error. This is difficult for sparse data. Our IL method from Chapter 2 does estimate the observed travel time on each link [62].

Combined with the assumption of constant speed across the link, this gives an estimate of the vehicle's position at all times. However, the model is complex and computationally challenging. The other alternative to estimating link travel times is to assume a travel model (i.e. a speed profile) that determines where the vehicle is on the path at all times. However, if the vehicle does not follow the travel model closely, then the inferred GPS errors can be inaccurate.

We use a Metropolis-within-Gibbs framework to estimate the posterior distribution over the missing data and unknown parameters, given the GPS data $\{G_i\}_{i=1}^I$. After initializing the unknowns, we iteratively update each unknown, conditional on the other unknowns, via Metropolis-Hastings sampling. The resulting Markov chain has state space $\left\{\{A_i, B_i\}_{i=1}^I, \mu_B, \mu_E\right\}$. First we describe how each unknown is initialized, and then describe how they are updated in the Markov chain.

4.3.2 Initializing the Path and GPS Parameters

First we describe how we initialize the missing data $\{A_i, B_i\}_{i=1}^I$ and the parameters μ_B and μ_E. The initial sample for the path A_i is actually quite important, because if the initial sample is very far from the GPS data, there may be a long transient period before the Markov chain is able to transition close to the GPS data, if it is able to transition there at all (see Section 4.3.3). Therefore, we initialize the path to be close to the GPS readings, using the following method.

1. Select every rth GPS reading to route the initial path through. The choice of r depends on the frequency of the GPS readings. The thinned GPS readings should not be so far apart that the local sampling cannot move be-

tween the initial path and the true path. However, using more GPS points also increases the initialization time and leads to more opportunities to map a point to the wrong links (for example if it has large location error). We use $r = 3$ for experiments.

2. Map each selected GPS reading to the s closest links. For each of these links, find the shortest distance path to each of the four closest links to the next selected GPS reading. Therefore there are s^2 shortest paths. Repeat for all adjacent (thinned) GPS readings and take the initial path to be the shortest-distance path in this new graph from the first GPS reading to the last GPS reading [37]. We use $s = 5$ for experiments. Alternatively, we could map each reading to all the links within a certain distance.

To initialize the parameters μ_B and μ_E, we calculate the mean distance of the GPS locations to the closest link in the road network, over the trip dataset. We initialize μ_B and μ_E both equal to this mean distance. We could also initialize μ_B and μ_E randomly, for example with mean equal to this distance. The initial value does not appear to be important in the estimation of these parameters. To initialize the observed bias $B_i = \{R_i, \theta_i\}$ for each trip i, we take $R_i \sim \text{Exp}(1/\mu_B)$, using the initialized value of μ_B, and take $\theta_i \sim \text{Unif}(0, 2\pi)$.

4.3.3 Updating the Paths

To update the path sample for trip i, we use a Metropolis-Hastings (M-H) proposal. The method used to propose a new path is the same as the one we introduced in Chapter 2, though the interpretation and acceptance rate are different, because the statistical model is different. Specifically, we uniformly choose an

node v_1 from the path A_i, excluding the final node. Let r be the number of nodes following v_1 in the path. We draw a random integer $j \sim \text{Unif}(1, \min\{r, K\})$ and denote the jth node following v_1 as v_2. We then collect all the routes between v_1 and v_2 in the road network of length at most K, of which the current route between v_1 and v_2 must be one, and uniformly propose a new route from this set to be a change to the path, giving proposed path A_i^*. We accept A_i^* as the new path with the appropriate M-H acceptance probability, which equals

$$p_{MH} = \frac{\mathcal{L}\left(A_i^*, B_i, G_i | \mu_B, \mu_E\right) \pi(A_i^*)}{\mathcal{L}\left(A_i, B_i, G_i | \mu_B, \mu_E\right) \pi(A_i)} \frac{N_i \min(r, K)}{N_i^* \min(r^*, K)}$$

where π and $\mathcal{L}$ are the prior and likelihood functions defined in Section 4.3.1, N_i is the number of links in the path A_i, and N_i^* and r^* are the corresponding values to N_i and r for the proposed path.

Since the start and end locations of the trip are unknown, we also must be able to update the estimated start and end nodes of the path. On its surface this proposal does not allow this, because only an interior portion of the path can be changed. However, the start and end nodes can be changed if we append a dummy start and end node to the beginning and end of the path, and connect these nodes with dummy links to all the nodes near to the first and last GPS readings. Then the above update on the path will allow the real start and end nodes to change between the nearby nodes to the first and last GPS readings. To select the set of nearby nodes, we take the five nearest links to the first and last GPS readings. Alternatively, we could use all the links within a certain distance from the first and last readings.

As we observed in Section 2.3.5, the Markov chain generated by this proposal is irreducible if it is possible to move between any possible paths between the (dummy) start and end node in a finite number of steps [62]. The road net-

work and the maximum update length K determine whether this is possible. It is advisable to keep K as small as possible, because the acceptance rates decrease with K. It may be impractical for K to be large enough for the Markov chain to be irreducible for all possible trips [62]. However, the decrease in acceptance rate as K grows is slower than in Chapter 2, because we do not have to update link travel times. When proposing many new link travel times in a single update in Chapter 2, the acceptance rate becomes low because often one (or more) link travel times is very unlikely in its distribution. We use $K = 10$ in this chapter, as opposed to $K = 6$ in Chapter 2.

Finally, we compare the proposal discussed here to the one introduced by Flötteröd and Bierlaire [14]. The proposals are similar, in that both change an interior portion of the path between two nodes v_1 and v_2. The proposals differ in that we restrict the number of links between v_1 and v_2 to be at most K and uniformly choose a new route to replace it. Flötteröd and Bierlaire also denote a current middle node v_3 between v_1 and v_2, and choose a new middle node v_3^* via a distribution on nodes in the road network, and route the proposed path through this new middle node, using fastest paths from v_1 to v_3^* and v_3^* to v_2. Since the reverse transition is impossible if the current routes from v_1 to v_3 and v_3 to v_2 are not fastest paths, they do not propose a new path in these cases.

Unlike in Flötteröd and Bierlaire's method, proposals are always possible in our method from any state. However, our Markov chain is reducible if the integer K is too small. Another difference is that we do not require any fastest paths to be precomputed and stored or computed at each iteration. Precomputing and storing fastest paths is memory-intensive if the road network is large, while computing fastest paths at every iteration can be computationally inten-

sive. Finally, Flötteröd and Bierlaire do not consider cases where the start and end locations of the vehicle are unknown.

4.3.4 Updating the GPS Error Parameters and Observations

Next we describe how we update the GPS bias vectors $B_i = \{R_i, \theta_i\}$ and the bias and error parameters μ_B and μ_E. Again we use Metropolis-Hastings proposals. For the bias magnitude R_i for trip i, we use a lognormal proposal $R_i^* = \mathcal{LN}(\log(R_i), \xi_B^2)$ with fixed variance ξ_B^2. We accept R_i^* with the appropriate M-H acceptance probability. The ratio of likelihoods in the M-H acceptance probability (see Equation 4.1) for the proposed and current states reduces to

$$\frac{\mathcal{L}(\, A_i, \{R_i^*, \theta_i\}, G_i |\, \mu_B, \mu_E)}{\mathcal{L}(\, A_i, \{R_i, \theta_i\}, G_i |\, \mu_B, \mu_E)} = \frac{\text{Exp}(R_i^*; 1/\mu_B) \prod_{\ell=1}^{m_i} \text{Exp}\left(D_i^{\ell *}; 1/\mu_E\right)}{\text{Exp}(R_i; 1/\mu_B) \prod_{\ell=1}^{m_i} \text{Exp}\left(D_i^{\ell}; 1/\mu_E\right)},$$

where the distance D_i^ℓ between the bias-removed GPS location $Z_i^\ell - B_i$ and the nearest point on the path also changes to become $D_i^{\ell *}$, since the bias changes.

To update the bias direction θ_i, we use a normal proposal modulo 2π, i.e. take $W_i \sim \mathcal{N}(\theta_i, \xi_\theta^2)$, where ξ_θ^2 is the proposal variance, and take $\theta_i^* = W_i \mod 2\pi$. The likelihood ratio reduces to

$$\frac{\mathcal{L}(\, A_i, \{R_i, \theta_i^*\}, G_i |\, \mu_B, \mu_E)}{\mathcal{L}(\, A_i, \{R_i, \theta_i\}, G_i |\, \mu_B, \mu_E)} = \frac{\prod_{\ell=1}^{m_i} \text{Exp}\left(D_i^{\ell *}; 1/\mu_E\right)}{\prod_{\ell=1}^{m_i} \text{Exp}\left(D_i^{\ell}; 1/\mu_E\right)},$$

where the proposed values $D_i^{\ell *}$ are determined by θ_i^*.

For the mean GPS bias magnitude μ_B and mean GPS remaining error μ_E, we also use lognormal proposals. For μ_B, the ratio of likelihoods reduces to

$$\frac{\mathcal{L}(\, \{A_i, B_i, G_i\}_{i=1}^I \Big|\, \mu_B^*, \mu_E)}{\mathcal{L}(\, \{A_i, B_i, G_i\}_{i=1}^I \Big|\, \mu_B, \mu_E)} = \frac{\prod_{i=1}^I \text{Exp}(R_i; 1/\mu_B^*)}{\prod_{i=1}^I \text{Exp}(R_i; 1/\mu_B)},$$

while for μ_E, the ratio of likelihoods reduces to

$$\frac{\mathcal{L}(\{A_i, B_i, G_i\}_{i=1}^{I} \Big| \mu_B, \mu_E^*)}{\mathcal{L}(\{A_i, B_i, G_i\}_{i=1}^{I} \Big| \mu_B, \mu_E)} = \frac{\prod_{i=1}^{I} \prod_{\ell=1}^{n_i} \text{Exp}(E_{i\ell}; 1/\mu_E^*)}{\prod_{i=1}^{I} \prod_{\ell=1}^{n_i} \text{Exp}(E_{i\ell}; 1/\mu_E)}.$$

The variances for the four proposals given in this section can be tuned to achieve a desired acceptance rate [52].

4.3.5 Fixing the Constants

Here we describe how we fix the constants C, p_{L}, p_{R}, $\{\{\sigma_{i,\ell}^2\}_{\ell=1}^{m_i-1}\}_{i=1}^{I}$, and $\{\tau(j)\}_{j=1}^{J}$. Recall that $\sigma_{i,\ell}^2$ is a function of the estimated distance Δd_i^ℓ between GPS readings ℓ and $\ell+1$ in trip i. In previous work, we observed that the variance of travel times on the log scale is larger for short trips than for long trips, and that the log scale variance follows a roughly exponential decay in the trip distance (Section 3.3.2). Although full trip travel times likely behave differently than portions of trips of the same length, because of speed-up and slow-down effects, it is reasonable to use the analysis for full trips in Toronto to estimate a value for $\sigma_{i,\ell}^2$ (travel time variability for a portion of the trip). We use the same exponential decay model as in Chapter 3, and obtain $\sigma_{i,\ell}^2 = Me^{-\nu D_i^\ell} + \delta$, where $M = 0.22$, $\nu = -0.0008$, $\delta = 0.08$. These values were calculated in Section 3.3.2 and are kept fixed for all experiments in this chapter.

The expected link travel times $\tau(j)$ can be taken from prior knowledge or a travel time estimation method. For example, given GPS speed data, the local travel time estimation methods introduced in Chapter 2 can be used to provide a straightforward estimate of $\tau(j)$. We use previous travel time estimates for each link provided by The Optima Corporation. We fix the turn penalties to

reasonable values for ambulance trips, namely $p_{\mathrm{L}} = 10$ seconds and $p_{\mathrm{R}} = 5$ seconds. We have experimented with higher turn penalties ($p_{\mathrm{L}} = 20$ seconds and $p_{\mathrm{R}} = 10$, for example) but found slightly worse results in general, although higher turn penalties could be helpful in certain cases (see Sections 4.4 and 4.5). Finally, we set C for each dataset according to the principle that for a trip of average duration, a path with typical travel time 10% longer should be 10 times less likely [62]. This yields values ranging from $C = 0.11$ and $C = 0.14$ for the various simulated and real datasets tested.

4.4 Toronto Ambulance Data Experiments

In this section and the next, we describe map-matching experiments on ambulance data from Toronto and on simulated data on the Toronto road network. We evaluate the performance of the map-matching method introduced in Section 4.3. We compare this method to a method where there is no GPS bias, only independent error, but with all other characteristics of the model the same. We refer to the method with both GPS bias and independent error as the full method, and the method with only independent error as the reduced method.

4.4.1 Toronto Data

In this section, we use data from ambulances in Toronto, collected from 2007-2008. There are 157,235 trips in this dataset, each consisting of a sequence of GPS observations. Unfortunately, we do not have ground truth paths traveled for these trips. Preprocessing for this dataset was discussed at length in Sec-

tion 3.3.1. We also define a 3x3 kilometer region of downtown Toronto for special study, because the GPS location error appears more severe there. There are 15,482 trips with at least one GPS reading in this downtown region.

First we highlight exploratory data analysis on the GPS errors in the Toronto dataset and the downtown region in particular. To obtain initial estimates of the distribution of GPS errors, we calculate the distance from each GPS location to the nearest link in the road network. The "Whole city" and "Downtown" rows of Table 4.2 show various quantiles of this distribution for the entire dataset and the downtown region.

Quantile	0.25	0.5	0.75	0.9	0.99	0.999
Whole city (m)	2.2	4.8	8.3	12.6	70.1	2193
Exp(1/6.41)	1.8	4.4	8.9	14.8	29.5	44.3
Downtown (m)	2.7	6.0	13.3	27.1	64.7	165.6
Exp(1/10.53)	3.0	7.3	14.6	24.2	48.5	72.6

Table 4.1: Quantiles of distributions of GPS distance (in meters) to nearest link, together with quantiles from related exponential distributions.

First, we observe that there are many extremely large GPS errors, especially in the non-downtown portion of the dataset. In order to estimate the GPS error distribution, it is probably best simply to remove trips that have extremely large errors, for example 500 meters (m) or more, because in these cases we probably will not be able to estimate the path correctly. The downtown region has larger error than the entire city in general, having median 6.0 m compared to 4.8 m, but not in the right tail. Most readings in both datasets have reasonably small distance; the 0.9 quantile is 12.5 m for the whole city and 26.7 m for downtown.

We also report quantiles for exponential distributions, where the mean of each exponential distribution equals the mean of the corresponding GPS dis-

tance distribution, truncated at 100 m. The exponential distributions are a good fit for the body of the distribution, but have lighter tails than the GPS distance distributions. As observed in Section 4.1, GPS errors are often assumed to be bivariate normal. The Rayleigh distribution is the magnitude of a symmetric bivariate normal distribution, but is a much worse fit to these data than the exponential. Assuming that the GPS errors are bivariate normal does not lead the closest-distance distribution to be Rayleigh in any case, because the closest distance is typically smaller than the error magnitude. In the case of a straight, infinitely long road, the closest distance equals one component of the bivariate normal error, i.e. a folded normal [62]. A folded normal distribution is also a worse fit to these data than the exponential.

4.4.2 Map-Matching Results

We now consider results of our full and reduced map-matching methods on the Toronto ambulance data. We sample 500 trips at random from the downtown region and the non-downtown region to be map-matched. We report posterior mean estimates for the full and reduced methods on these samples in Table 4.2.

	Full model		Reduced model
Dataset	μ_B	μ_E	μ_E
Downtown trips	32.7	8.3	17.9
Non-downtown trips	15.6	5.4	9.0

Table 4.2: Posterior mean parameter estimates from the full and reduced map-matching models on Toronto sample datasets.

First we discuss a potential issue of the information-sharing for estimating the GPS distribution error used by our methods. For the results in Table 4.2,

we removed three trips from the non-downtown dataset that our method was not able to map-match correctly. If we do not remove these three trips, the parameter estimates from the full model for the non-downtown dataset are much larger: $\mu_B = 18.7$ m and $\mu_E = 17.2$ m, because the three trips have very large inferred GPS errors. In cases where the map-matching fails, typically either the trip actually does have very high GPS error, or our initialization method (Section 4.3.2) fails to find a path reasonably close to the GPS readings.

To protect against this undesirable behavior, it is advisable first to remove trips with very high GPS errors, and also to evaluate the results of the Markov chain to find other trips with very poor map-matching estimates. These trips typically have very high estimated biases or estimated remaining independent GPS errors. Using a heavy-tailed distribution in our model instead of an exponential could also mitigate this issue, and is an area of current work.

From the estimates of μ_B and μ_E in Table 4.2, the full model appears to assign the bulk of the GPS error to be bias, rather than the independent error. We are interested in whether this is a true feature of the data or an artifact of our model. To test this, we use the map-matching estimates from the reduced model, so there is no direct preference for estimated paths that include GPS bias, and calculate the signed distance from each GPS location to the nearest link in the estimated path, with the restriction that the GPS readings must occur in their observed sequence. The sign of the distance refers to whether the GPS location is to the right or left of the estimated path. We use signed distance so that GPS locations on opposite sides of the road are known to have different biases. For this analysis, we ignore the first and last GPS readings in each trip, because the ends of the trip are typically the most difficult to map-match (see below), and

therefore can give misleading information about the GPS bias.

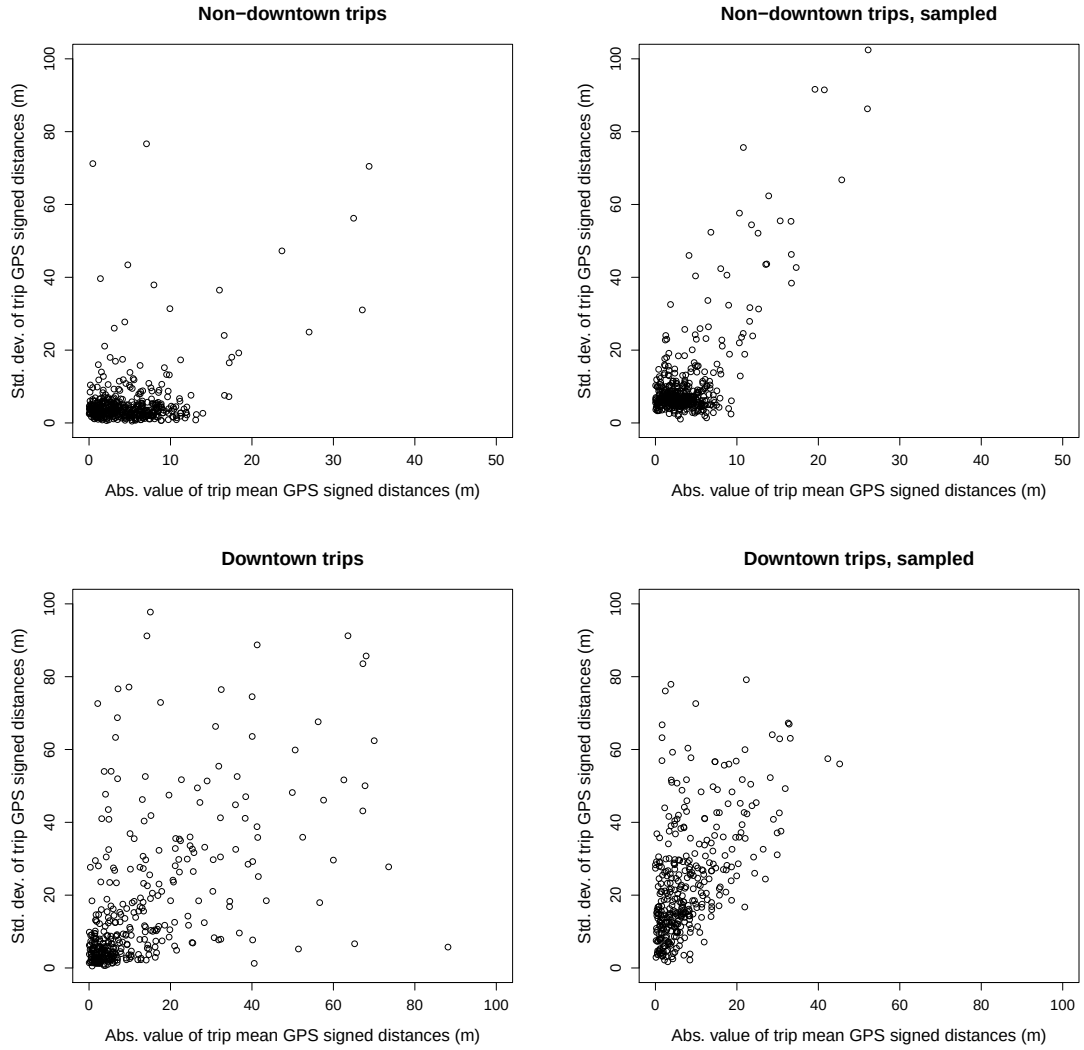

Figure 4.2: Scatterplots of the absolute value of the trip mean GPS signed distances vs. the trip signed distance standard deviations. Top: Non-downtown Toronto, Bottom: Downtown Toronto. Left: Values from the true trip data, Right: The same trips with randomly sampled GPS errors.

In Figure 4.2, we plot the absolute value of the mean signed distance for each trip on the x-axis vs. the standard deviation of the signed distances for each trip on the y-axis. The top plots are from the non-downtown dataset and the bottom plots are from the downtown dataset. The left plots show the values as

calculated in the previous paragraph. The right plots show values for sampled GPS errors for each trip. For trip i, having m_i GPS readings, we independently sample $m_i - 2$ values from the empirical distribution of signed GPS distances for the whole dataset (subtracting 2 to be consistent with ignoring the first and last readings), and calculate the absolute value of the mean and the standard deviation for these samples. This illustrates the relation between mean signed distance and standard deviation if the GPS errors are independent. A few outliers are left off of each plot.

The figure for the downtown true data is quite different than the figure for the downtown sampled data. The observed trip standard deviations are typically lower in the true data than in the sampled data; the median true s.d. is 8.3 m while the median sampled s.d. is 21.9 m. Thus the errors typically have lower variability across a given trip than they would if drawn independently. The figures for the non-downtown trips appear more similar, but again the true standard deviations are smaller in general than the sampled values; the median true s.d. is 3.6 m while the median sampled s.d. is 6.8 m. This is evidence that the trips in the real data do in fact show persistent GPS bias.

It is also interesting to note that the practice of driving on the right side of the road is apparent in the non-downtown dataset. Driving on the right corresponds to a positive signed distance, since the distance is calculated to the road centerline, and the median signed distance for the non-downtown dataset is 2.8 m. However, the median signed distance for the downtown data is only 0.9 m. There may be more trips in the non-downtown region on multi-lane roads where the bias induced by driving on the right is large. We observe similar results with other larger sets of trips from the Toronto dataset, so we do not

believe that this is an artifact of the dataset samples used here.

Finally, we consider six example trips from the downtown dataset, and analyze the differences in their map-matching estimates from the full model and the reduced model. The trips are chosen for their interest and therefore have larger GPS location error than many other trips. However they are typical in that the location error appears to be predominately persistent bias. We have seen almost no examples of trips in either the downtown or non-downtown dataset where the independent error appears to be more substantial than the bias.

The six example trips are shown in Figure 4.3. The first GPS reading is shown as a solid circle and the rest as open circles. The estimated path from the full model is shown as a solid line, and the estimated path from the reduced model as a dashed line. As mentioned above, most of the GPS readings are separated by 200 m, although sometimes the distance is larger or smaller than this, for example in the top-left and bottom-left figures.

These trips show three situations where the full model appears to outperform the reduced model. First, there are cases where the reduced model takes a route that is closer to some GPS locations, but is longer in expected travel time or requires more turns than the route taken by the full model. Second, there are cases where the reduced model takes a route that is farther away from some GPS readings than the full model, but is shorter and perhaps uses fewer turns. Third, the full model often appears to estimate the beginning or end of the path more effectively.

The top two paths are examples of the first situation. The reduced model takes a detour or extra turns to get closer to the GPS readings, and the paths

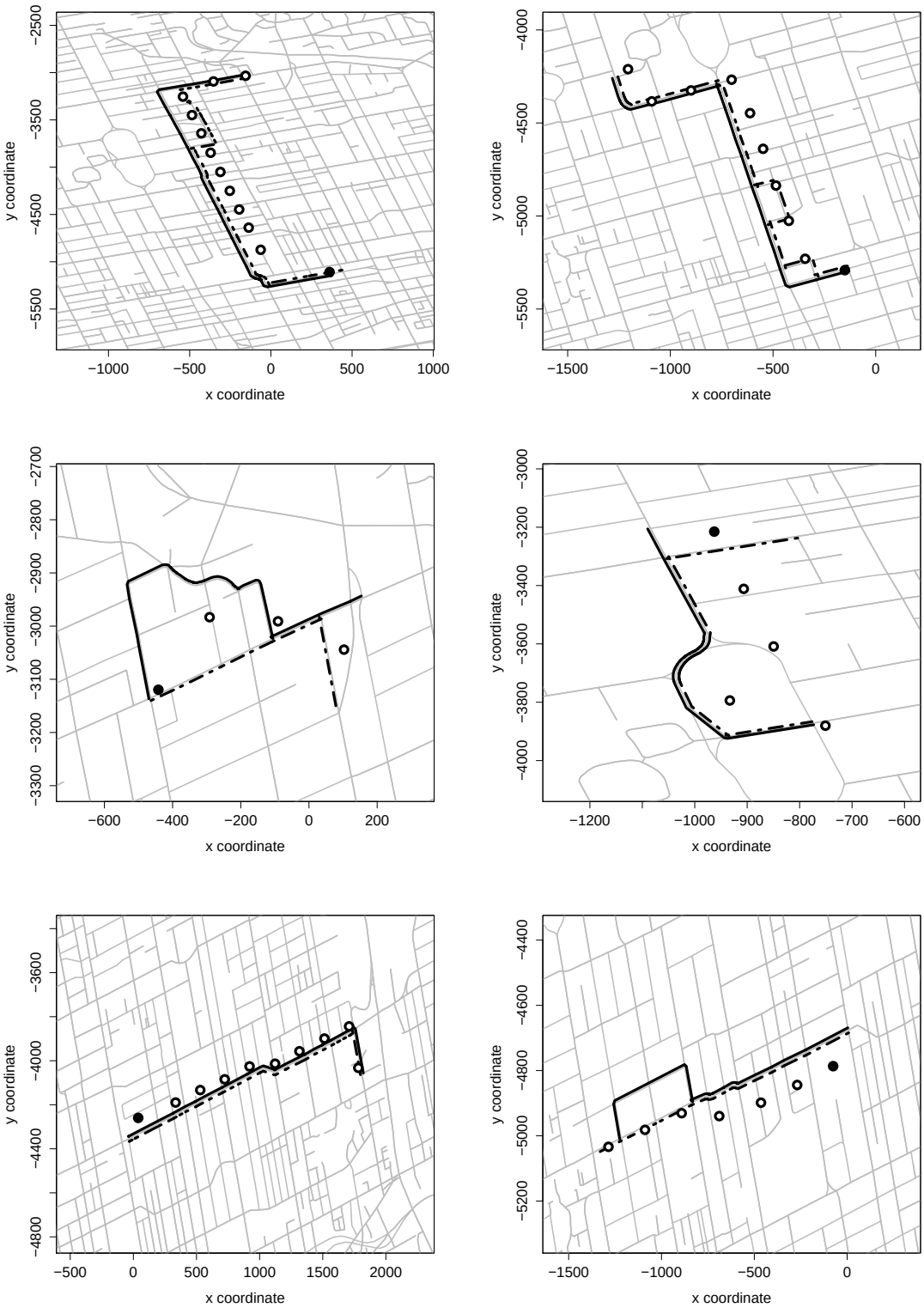

Figure 4.3: Six example ambulance paths from the downtown Toronto dataset.

from the reduced model appear incorrect. These two paths could likely be estimated correctly by alternative map-matching techniques to our full model. For example, larger turn penalties would give less incentive for the reduced model to make these extra turns, and a stronger preference towards short trips would discourage the detour. However, these modifications would hinder the reduced model in the second situation. The middle-left path is an example of this situation. It is not perfectly clear what the true route is. However, the full model finds a path where the bias appears to be very persistent. The path taken is an odd one, but there are other examples in the dataset where the ambulance clearly turns around or takes other odd paths. We see more examples of the second situation in the simulated data of Section 4.5.

These first two situations are opposites. The reduced model appears to make some errors choosing paths that are too long and close to the GPS readings and other errors choosing paths that are too short and far from the GPS readings. Therefore, it appears that tuning the parameters of the distributions assumed by the reduced model for GPS location error and the multinomial logit choice model prior, i.e. adjusting the tradeoff between short paths and paths closer to the GPS readings, will not be able to fix all errors. Modeling the bias directly can therefore be useful.

The middle-right path is an example of the third situation. Near the beginning of the path, the estimated bias allows the full method to choose a route that is farther from the first GPS reading than the route chosen by the reduced method, but which appears to be correct, given the very persistent bias. This is a common situation. The beginning and end of the path are typically more difficult to estimate than the middle, because the constraints imposed by the

other GPS locations in the path are only on one side, and therefore there can be multiple reasonable routes. Using GPS heading information could be an alternative way to distinguish the correct path in some examples like this, but this information is not always available [41].

In the bottom-left path, even though there is persistent bias, both models appear to estimate the path correctly. This is another common situation. In the bottom-right path, the reduced model appears to estimate the path correctly, while the full model does not. The GPS bias appears to change dramatically after the fourth GPS reading, becoming much smaller. Attempting to maintain a more constant bias, the full model takes an incorrect detour. Examples of this type are rare, but they can lead to map-matching errors by the full model.

4.5 Simulated Data Experiments

In this section, we describe results with simulated data on the downtown Toronto road network. These experiments are useful because there is ground truth path data, which allows us to assess map-matching performance, and also because we are able to vary the characteristics of the GPS location error, to compare the full method and the reduced method in a range of settings. There are two other frameworks for testing map-matching methods that have been used in the literature. The first is to use a set of real ambulance trips where the true paths are known by a non-GPS method. Typically these datasets are generated specifically for the map-matching experiment, and therefore are fairly small [46]. A second framework is to take GPS data from trips where the true paths are not known, and manually map-match them [41]. Depending on the

magnitude of GPS error, this may not be possible to do perfectly.

4.5.1 Generating Simulated Data

First we describe how our simulated data is generated. To use as the true path for each trip, we use a map-matching estimate from a variant of our reduced method for a trip from the real Toronto ambulance dataset. We use a different set of 500 trips from the downtown Toronto dataset than used in Section 4.4.2. Using a map-matching estimate as the true path ensures that the simulated path is fairly close to a real path traveled by a vehicle. It is important to make the simulated paths realistic, because map-matching performance may depend on the shape of the paths.

Given the simulated path for trip i, we draw a lognormal random variable to be the true trip duration T_i. The model we use to do this is the one introduced in Chapter 3 (Section 3.2.1). The mean and variance of the lognormal distribution depend on the distance traversed on each link, on estimated speeds for each road class, and other parameters. We use the estimated parameter values from the results in Table 3.1.

To simulate GPS data, we need the location of the vehicle at each time. We achieve this by drawing link travel times, given the trip travel time T_i, and assuming that the vehicle moves at constant speed across each link. We use a Dirichlet distribution to distribute the total travel time across the links in the path. Specifically, again denote the true path as $A_i = \left\{A_i^1, \ldots, A_i^{N_i}\right\}$ and let $\tau(A_i^j)$ be the expected travel time on link A_i^j. We obtain $\tau(A_i^j)$ from the travel time model in Section 3.2.1. Define $f(j) = \tau(A_i^j)/\sum_{j=1}^{N_i} \tau(A_i^j)$ to be

the fraction of the expected trip travel time on link A_i^j. Then draw a vector $p \sim \text{Dirichlet}(\eta f(1), \ldots, \eta f(N_i))$, where η is a constant, and set the travel time for each link A_i^j equal to $p_j T_i$. This gives expected travel time for link A_i^j equal to $\tau(A_i^j)$, because $E(p_j) = f(j)$. The constant η controls the variances of the link travel times; a larger η gives smaller variances [16]. We set $\eta = 50$.

Given the path traversed and link travel times, we generate simulated GPS observations using the same procedure as in Chapter 2 (Section 2.6.1), which is to sample GPS readings at fixed travel distances along the path. Specifically, we draw a new GPS reading every time the vehicle travels 250 meters (m). This value is used because GPS readings in the Toronto data are typically separated by 200 m in straight-line distance. Our simulated GPS readings are somewhat sparser in general than the data used by Bierlaire, Chen, and Newman [5], which was recorded every 10 seconds, because 250 m in 10 seconds would correspond to 56 miles per hour. The time of the GPS reading is the time at which the vehicle was at the corresponding location.

The GPS readings have location error. We vary the characteristics of the simulated location error widely, to compare our full and reduced models in a range of situations. We intentionally use distributions for the GPS error that do not match the distributions assumed in our model. Some of the datasets have larger error than appears to be common in the real Toronto data. However, there are some trips with very high error in the Toronto data, so we wish to assess map-matching performance on a large number of trips of this type.

We consider four types of location error. In the first type, there is only GPS bias, not independent error. The bias is simulated by drawing a uniform random direction $\theta \sim \text{Unif}(0, 2\pi)$ and a uniform random magnitude

$R \sim \text{Unif}(0, M)$. We create four different datasets by varying the maximum M, taking $M \in \{20, 50, 100, 200\}$ meters. In the second dataset type, there is both GPS bias and independent error. The GPS bias again contains a uniform direction and magnitude, but the independent error is drawn from a bivariate normal distribution, $N(0, \Sigma)$, where $\Sigma = \left(\begin{smallmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{smallmatrix}\right)$. We make three different datasets, each with large bias ($M = 100$) and independent error varying from small to large, taking $\sigma \in \{8, 20, 50\}$ meters. In the third dataset type, there is only independent error, again drawn from a bivariate normal. We report results from one dataset of this type, with $\sigma = 50$ m.

In the fourth dataset type, again there is only GPS bias, but the bias is allowed to change in the middle of the trip. For each reading in the trip, beginning with the first reading, we draw a new bias vector (both magnitude and direction) with probability p and keep the previous bias vector with probability $1 - p$. Therefore, the number of consecutive readings with the same bias follows a geometric distribution with mean $(1 - p)/p$. We report results for two datasets of this type, both with $p = 0.2$, and with $M \in \{50, 200\}$. These datasets mimic the situation we occasionally see in real trips, where the bias appears to shift in the middle of the trip, as in the bottom-right example in Figure 4.3.

4.5.2 Map-Matching Results

Here we discuss results of the full and reduced models on the simulated datasets introduced in Section 4.5.1. First we report posterior mean estimates for the parameters μ_B and μ_E for the full model and μ_E for the reduced model, shown in Table 4.3. The rows of the table correspond to the ten simulated datasets

with varying GPS location error introduced above. The first four rows are the datasets with only bias, the next three rows the datasets with both bias and independent error, the eighth row the dataset with only independent error, and the final two rows the datasets with bias that can change at each reading.

	Full model		Reduced model
GPS error distribution	μ_B	μ_E	μ_E
Unif(0,20)	13.9	0.2	7.1
Unif(0,50)	29.7	1.5	29.7
Unif(0,100)	51.8	2.8	35.7
Unif(0,200)	86.5	28.0	73.8
Unif(0,100) + N(0,8)	49.5	8.4	37.8
Unif(0,100) + N(0,20)	49.9	18.4	47.6
Unif(0,100) + N(0,50)	59.1	54.3	62.8
N(0,50)	49.0	56.0	56.6
Unif(0,50), p=0.2	26.4	10.9	19.0
Unif(0,200), p=0.2	72.8	53.3	72.2

Table 4.3: Parameter estimates from the full and reduced map-matching methods on simulated datasets.

In the first three datasets, the full model is able to determine correctly that the location error is predominantly bias, since the estimates for μ_E are very small. In the fourth dataset, which has very large bias, the full model incorrectly estimates a fairly large independent error distribution ($\mu_E = 28.0$), but the bias magnitude distribution is still larger. The estimated mean biases for these four datasets are also roughly correct; since the simulated bias magnitude is Unif$(0, M)$, the mean is $M/2$, which is close to the estimate of μ_B in each case.

The reduced method also estimates the mean error of the first four datasets fairly well. Since the independent error is measured from the GPS location to the closest point on the path, it is smaller than the true error magnitude. How much smaller depends on the curvature of the road. For an infinitely long, straight road, the distance to the nearest point corresponds to one component of the

two-dimensional error [62]. For error with an independent, uniform angle, the mean of one component of the error is $2/\pi$ times the mean magnitude, equaling $0.637M/2$ or $\{6.4, 15.9, 31.8, 63.7\}$ for the first four datasets.

In the next three datasets, there is large bias and a range of independent error from small to large. Again the full model is able to recover the relative magnitudes of GPS bias and independent error reasonably correctly. In the eight dataset, where there is only large independent error, the full model is not able to identify that there is no bias, but gives both μ_B and μ_E large values. In the final two datasets, the full method similarly gives μ_B and μ_E fairly large values.

Next we assess map-matching performance of the two methods on each simulated dataset. We use the true positive and false positive rates (TPR and FPR) introduced by Rahmani et al. [46]. These are:

$$\text{TPR} = \frac{d(\text{Est}_i \cap \text{True}_i)}{d(\text{True}_i)}, \qquad \text{FPR} = \frac{d(\text{Est}_i - \text{True}_i)}{d(\text{True}_i)},$$

where $d(S)$ denotes the total length (distance) of a set of links S, Est_i is the estimated path for trip i, and True_i is the true path for trip i. Together these measures provide a good evaluation of map-matching performance, whereas one measure by itself might give an incomplete view [46]. For example, a method that assigns every path to take all links would trivially obtain 1 for the true positive rate, but would also have very high false positive rate. Alternatively, the number of links can be used instead of the distance. We also calculated false and true positive rates for only the interior links in the path, since the beginning and end of the path are typically the most difficult to estimate. Both methods perform better on the interior links than on the whole path. However, comparisons between the two methods are similar with either of these changes, so we only report the standard error rates defined in Equation 4.5.2.

	Full model		Reduced model	
GPS error distribution	TPR	FPR	TPR	FPR
Unif(0,20)	0.935	0.017	0.934	0.012
Unif(0,50)	0.922	0.029	0.884	0.037
Unif(0,100)	0.904	0.050	0.844	0.070
Unif(0,200)	0.739	0.184	0.677	0.202
Unif(0,100) + N(0,8)	0.889	0.062	0.841	0.078
Unif(0,100) + N(0,20)	0.873	0.055	0.824	0.066
Unif(0,100) + N(0,50)	0.786	0.110	0.788	0.103
N(0,50)	0.789	0.093	0.820	0.073
Unif(0,50), p=0.2	0.911	0.031	0.909	0.025
Unif(0,200), p=0.2	0.736	0.162	0.734	0.148

Table 4.4: Map-matching error rates on simulated datasets.

The error rate results are given in Table 4.4. Both methods perform well on the first dataset, where there is small bias and no independent error. For the other three datasets with only bias, the full model performs substantially better in TPR (4-6% higher) and slightly better in FPR. In the fifth and sixth datasets, with large bias and small-to-medium independent errors, the full model still performs substantially better in TPR (5% higher).

On the other hand, the full model performs worse on the eighth dataset (3% lower TPR and 2% higher FPR), where there is only large independent error. The two models perform comparably on the dataset with large bias and large independent error (the seventh dataset), and on the datasets where the bias can change (the ninth and tenth datasets). Extending the model to allow the bias to change during the path is an interesting area for further research.

Finally, in Figure 4.4 we examine six example paths from the third simulated dataset, which has Unif$(0, 100)$ bias magnitude and no independent error. As in the examples in Section 4.4, the path estimated by the full model is shown as a solid line and the path estimated by the reduced model as a dashed line. The

first GPS reading is again shown by a solid circle and the rest by open circles. The true path in the simulated data is now shown by a wide gray line.

We see similar behavior in these figures to the paths from real data analyzed in Section 4.4. The same three situations where the full model outperforms the reduced model arise: (1) where the reduced model takes a longer (in expected time) route to get closer to the GPS readings, (2) where the reduced model take a shorter route farther away from the GPS readings, and (3) where the full model is more successful in estimating the beginning and end of the path.

The top-left figure is an example of the second and third situations. The reduced method travels straight instead of turning at the beginning of the path, and also takes an incorrect turn that shortens the end of the path but is farther from the second-to-last GPS reading. The full model makes a mistake at the end of the path, but estimates the rest correctly. In the top-right figure, the reduced method again takes a shorter route that is farther from the GPS readings. In both figures, the reduced method ends the path earlier than it should, also to shorten the path. This is a common situation. Even if the path continued for another link, the inferred error would be fairly high, and so shortening the path appears to be more desirable. Only the portion of the final link up to the closest point to the final GPS reading would be counted in expected travel time calculations (see Equation 4.3.1), but this still adds a reasonable amount to the path.

The middle-left figure shows an interesting case. The reduced model makes one major mistake, taking a horizontal route that is closer than the true route to the third-to-last GPS reading. Only a knowledge of the persistent bias allows the full model to estimate this path correctly. It is difficult to think of an alternative type of map-matching method that would achieve this, without incorporating

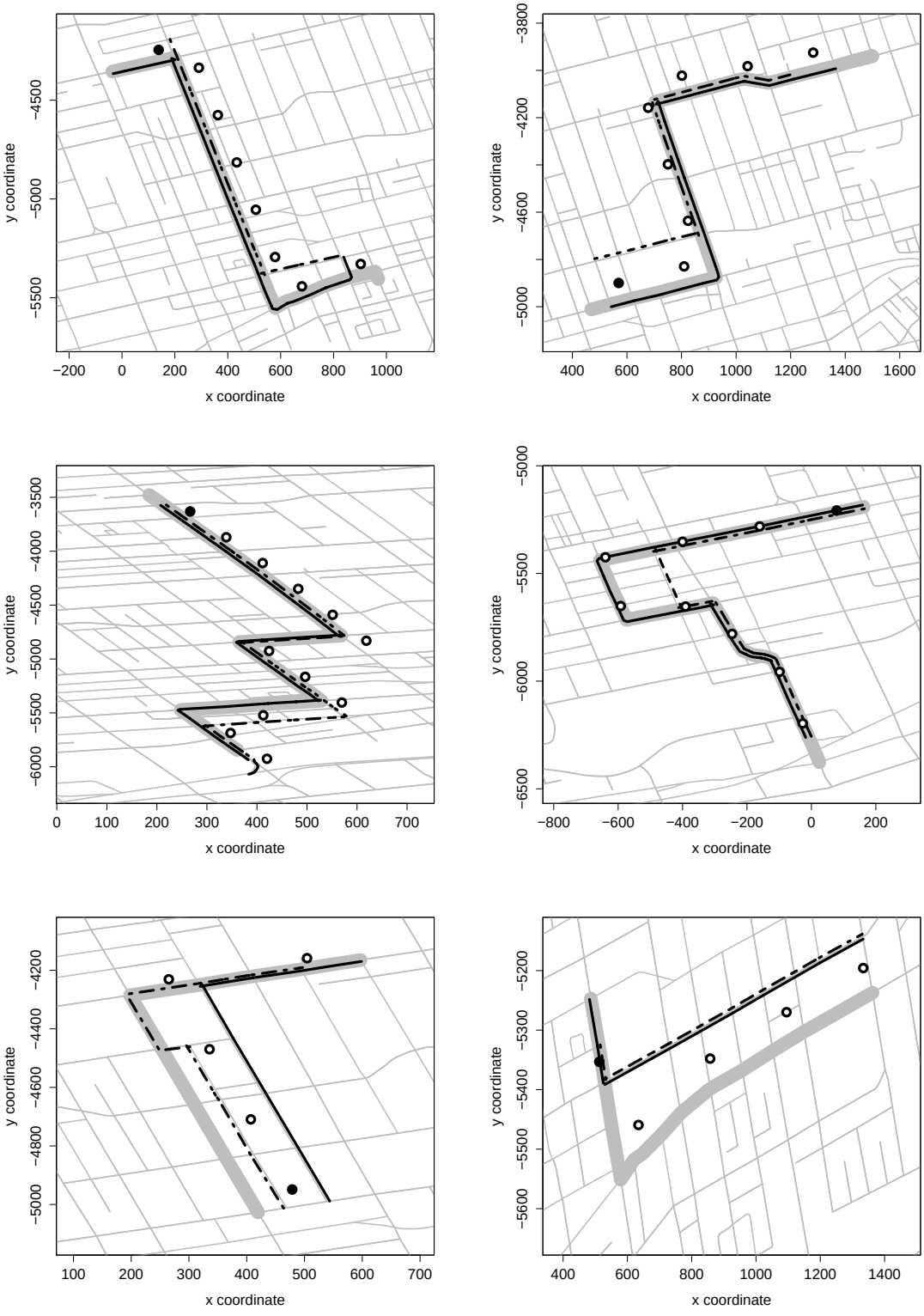

Figure 4.4: Six example paths from the simulated dataset with bias magnitude Unif(0,100).

dependence between the GPS location errors.

In the middle-right figure, the ambulance turns around. This is not a very common situation in real data, but does occur. The reduced method turns much earlier than it should. This error arises because the independent exponential error distribution is quite large. Unfortunately, some large errors that are incorrect may be tolerated. In some cases, a sharp density like the exponential (i.e. giving very small GPS errors a high density compared to larger errors) mitigates this behavior, but it does not in this case. It would be interesting to assess the effect of using a heavy-tailed distribution on this behavior.

The bottom two figures are examples of the identifiability issues discussed in Section 4.2. In the bottom-left figure, the path chosen by the full method is a translation of the true path, so the path and bias are not identifiable. The full model chooses the incorrect path because it leads to slightly smaller bias. This figure is also an example of the first situation above, because the reduced method takes extra turns to get closer to the GPS readings. The bottom-right figure is slightly different. Because of the odd angles in the true path, there is no alternative path that is a translation of the true path, and so the path and bias are identifiable. However, both models incorrectly estimate a shorter path. Given the path estimated by the full model, the second GPS reading is inferred to have some independent error. This is acceptable to the full method because its model allows both bias and independent error.

4.6 Conclusions

We considered the problem of map-matching sparse and error-prone GPS data with a persistent bias in GPS locations for all GPS readings in a trip, and potentially also independent location errors for each GPS reading. We observed from empirical evidence that persistent bias is a major component of GPS location error for ambulance trips recorded in Toronto.

First we considered whether the vehicle path and GPS location bias are identifiable, i.e. whether they can be uniquely determined given sufficient data. For the case with only unchanging bias and no independent error, we showed that the path and bias are identifiable up to translations of the path by a vector.

We introduced a statistical map-matching method where the GPS location error is modeled with an unchanging bias for the entire trip plus an additional independent error for each reading. We used a Bayesian model and computational method to estimate the paths traveled for an entire dataset of trips and the parameters of the GPS error distributions simultaneously. We tested our map-matching method on the data from ambulances in Toronto. We compared the full model with both GPS location error types to a reduced method with only independent location error between GPS readings. We found that the full model provided more realistic path estimates for three different types of example trips.

We also compared the two models on realistic simulated datasets of trips with a wide variety of GPS location error characteristics. We calculated true and false positive rates, in terms of fraction of path length correctly estimated, for map-matching performance on the simulated data. We found that the full model outperformed the reduced model by 4-6% in true positive rate on datasets where

the GPS bias was medium-to-large and the independent error was zero-to-medium. The two models performed comparably on datasets with low bias and independent error and on datasets with bias that was allowed to change in the middle of the trip. The reduced model performed slightly better on a dataset with only large independent error.

We are currently investigating using a heavy-tailed distribution for the GPS location bias and independent errors, which could more accurately match the empirical distribution of GPS errors, and could mitigate other issues arising in this chapter. It would also be interesting to investigate models allowing the bias to change during the trip. This appears to happen only in a small fraction of real trips, but when it occurs it can lead to map-matching errors from our model.

CHAPTER 5

CONCLUSIONS

In this chapter, we draw overall conclusions and consider areas for further work. We introduced two statistical methods for estimating vehicle travel time distributions, using Global Positioning System (GPS) data recorded during historical vehicle trips. In Chapter 2, we introduced our Independent Link (IL) method, using a model of the path taken by each vehicle in the data, the travel time on each link (road segment) in the network, and the GPS location and speed errors. We assumed independence between link travel times, and estimated the parameters of the model via a Markov chain Monte Carlo method. We compared the performance of the IL method with two simpler local methods and a recently published method from Budge et al. [8], using simulated data and data from ambulances in Toronto, on a subregion of Toronto. We found that the IL method outperformed the alternative methods in travel time point estimation. However, its interval estimates appeared unrealistically narrow.

In Chapter 3, we introduced our Whole Trip (WT) estimation method, using a model of the entire travel time of each trip, and including covariates such as the types of roads used and time of day. We again estimated the parameters of the model via a Markov chain Monte Carlo method. Modeling at the trip level allowed us to capture dependence between link travel times. The WT method also included fewer parameters and was more computationally efficient than the IL method. However, the WT model did lose some information compared to the IL model, because it ignored the interior GPS readings in each trip, once the path taken by the vehicle was estimated as a model input.

We compared the performance of the WT method with the method of Budge

et al. and commercially available travel time estimates from TomTom, using a large dataset of ambulance trips on the road network of Toronto. We found that the WT method outperformed the alternative methods in point and distribution estimation of travel times. We also found that the WT method outperformed the IL method in distribution estimation and was comparable in point estimation.

We also compared the WT method and the method of Budge et al. in their effect on ambulance management decisions, using a set of representative ambulance posts in Toronto. For each intersection in Toronto, we calculated which post was estimated to be the closest, according to the two methods. The two methods differed on closest posts for 5% of the intersections in the city, and so the methods could lead to different ambulance dispatch decisions for emergencies at those intersections. The two methods also differed substantially in estimating the probabilities an ambulance is able to reach each intersection in the city within a time threshold, responding from the closest post.

We also considered the map-matching problem, i.e. estimating a vehicle's path from a sequence of GPS readings. Our IL method simultaneously estimated map-matching solutions along with travel time distributions, and showed robustness to sparsity and locations errors in GPS readings. Our WT method required map-matching estimates for each historical vehicle path as inputs. In Chapter 4, we introduced a statistical map-matching method, motivated by the observation that successive GPS readings tend to exhibit persistent location biases. We observed that this method outperformed an alternative method that did not model GPS location bias, using simulated data with location bias and example trips from the Toronto ambulance data.

Finally, we discuss possible future research directions. First, it would be

interesting to explore extensions to the WT model, for example using semi-parametric methods. The assumption that the baseline travel time is a sum of link travel times plus an intercept (Equation 3.1 of Section 3.2.1) could be relaxed. Also, the trip effect could interact with the unit travel times for each link. For example, this could allow the time of day parameters to vary for different road classes or regions of a city.

Second, we observed that the WT method outperformed the IL method in distribution estimation for trip travel times, because the IL method's assumption of independence between link travel times led to unrealistically narrow travel time intervals. However, the WT method ignored the interior GPS readings in each trip. To use the interior GPS readings, we need a model of the movement of each vehicle during the trip. A possible extension is to model the trip travel time and also the link travel times, conditional on the trip travel time. For example, the trip travel time could be modeled by a lognormal distribution, which could be distributed across the links in the path by a Dirichlet distribution. Because such a model would need to estimate realized link travel times for each historical trip, it is likely to be computationally difficult. However, it would be interesting to compare its performance with the IL and WT methods.

Third, there is a wealth of real-time traffic information that is currently collected via smartphones and other navigation devices. For example, TomTom generates travel time predictions using both historical and real-time data. We tested TomTom's real-time predictions of travel times, but found that they did not perform as well as their historical data for estimating travel times for the Toronto ambulance data. This is not surprising, since the Toronto ambulance trips are historical. We should not expect real-time information from 2013 to

have any benefit for predicting historical data, even if the data are from the same time of day, for example.

However, EMS organizations make many real-time decisions about ambulance fleet management [11, 38], and real-time data could potentially be very useful for making these decisions. The real-time data would likely be for standard speed vehicles, not ambulances, because real-time data depends on traffic conditions at a very detailed level, and ambulance trips are comparatively rare. Thus there is again the difficulty that travel time distributions for these two cases are quite different, as we discussed in Chapter 3. However, it may be possible to combine historical ambulance data, historical standard speed data, and real-time standard speed data to obtain more accurate travel time estimates than can be made from historical ambulance data alone.

BIBLIOGRAPHY

[1] K. Aladdini. EMS response time models: A case study and analysis for the region of Waterloo. Master's thesis, University of Waterloo, 2010.

[2] R. Alanis, A. Ingolfsson, and B. Kolfal. A Markov Chain model for an EMS system with repositioning. *Production and Operations Management*, 22:216–231, 2012.

[3] M. Bernard, J. Hackney, and K.W. Axhausen. Correlation of link travel speeds. In *6th Swiss Transport Research Conference*. Ascona, Switzerland, 2006.

[4] P.J. Bickel and K.A. Doksum. *Mathematical Statistics, Volume I*. Prentice Hall, Englewood Cliffs, NJ, 2001.

[5] M. Bierlaire, J. Chen, and J. Newman. A probabilistic map matching method for smartphone GPS data. *Transportation Research Part C*, 26:78–98, 2013.

[6] N.E. Breslow and X. Lin. Bias correction in generalised linear mixed models with a single component of dispersion. *Biometrika*, 82:81–91, 1995.

[7] L. Brotcorne, G. Laporte, and F. Semet. Ambulance location and relocation models. *European Journal of Operational Research*, 147:451–463, 2003.

[8] S. Budge, A. Ingolfsson, and D. Zerom. Empirical analysis of ambulance travel times: The case of Calgary emergency medical services. *Management Science*, 56:716–723, 2010.

[9] W. Chen, Z. Li, M. Yu, and Y. Chen. Effects of sensor errors on the performance of map matching. *The Journal of Navigation*, 58:273–282, 2005.

[10] C. de Fabritiis, R. Ragona, and G. Valenti. Traffic estimation and prediction based on real time floating car data. In *11th International IEEE Conference on Intelligent Transportation Systems*, pages 197–203. IEEE, 2008.

[11] S.F. Dean. Why the closest ambulance cannot be dispatched in an urban emergency medical services system. *Prehospital and Disaster Medicine*, 23:161–165, 2008.

[12] E. Erkut, A. Ingolfsson, and G. Erdoğan. Ambulance location for maximum survival. *Naval Research Logistics (NRL)*, 55:42–58, 2008.

[13] J.J. Fitch. *Prehospital Care Administration: Issues, Readings, Cases*. Mosby-Year Book, St. Louis, 1995.

[14] G. Flötteröd and M. Bierlaire. Metropolis-Hastings sampling of paths. *Transportation Research Part B*, 48:53–66, 2013.

[15] A. Gelman. Prior distributions for variance parameters in hierarchical models. *Bayesian Analysis*, 1:515–533, 2006.

[16] A. Gelman, J.B. Carlin, H.S. Stern, and D.B. Rubin. *Bayesian Data Analysis*. Chapman & Hall, London, 2004.

[17] A. Gelman and D.B. Rubin. Inference from iterative simulation using multiple sequences. *Statistical Science*, 7:457–472, 1992.

[18] T. Gneiting, F. Balabdaoui, and A.E. Raftery. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society: Series B*, 69:243–268, 2007.

[19] T. Gneiting and A.E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102:359–378, 2007.

[20] J.B. Goldberg. Operations research models for the deployment of emergency services vehicles. *EMS Management Journal*, 1:20–39, 2004.

[21] P.J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82:711–732, 1995.

[22] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning, Second Edition*. Springer, New York, 2005.

[23] S.G. Henderson. Operations research tools for addressing current challenges in emergency medical services. In *Wiley Encyclopedia of Operations Research and Management Science.* Wiley, New York, 2010.

[24] A. Hofleitner, R. Herring, P. Abbeel, and A. Bayen. Learning the dynamics of arterial traffic from probe data using a dynamic Bayesian network. *IEEE Transactions on Intelligent Transportation Systems*, 13:1679–1693, 2012.

[25] A. Hofleitner, R. Herring, and A. Bayen. Arterial travel time forecast with streaming data: A hybrid approach of flow modeling and machine learning. *Transportation Research Part B*, 46:1097–1122, 2012.

[26] T. Hunter, P. Abbeel, and A.M. Bayen. The path inference filter: model-based low-latency map matching of probe vehicle data. In *Algorithmic Foundations of Robotics X*, pages 591–607. Springer, New York, 2013.

[27] A. Ingolfsson, S. Budge, and E. Erkut. Optimal ambulance location with random delays and travel times. *Health Care Management Science*, 11:262–274, 2008.

[28] E. Jenelius and H.N. Koutsopoulos. Travel time estimation for urban road networks using low frequency probe vehicle data. *Transportation Research Part B*, to appear, 2013.

[29] K.H.F. Kan, R.M. Reesor, T. Whitehead, and M. Davison. Correcting the bias in Monte Carlo estimators of American-style option values. *Monte Carlo and Quasi-Monte Carlo Methods 2008*, pages 439–454, 2009.

[30] W.D. Kelton and A.M. Law. *Simulation Modeling and Analysis*. McGraw Hill, Boston, 2000.

[31] W. Kim, G.I. Jee, and J. Lee. Efficient use of digital road map in various positioning for its. In *Position Location and Navigation Symposium, IEEE 2000*, pages 170–176. IEEE, 2000.

[32] P. Kolesar, W. Walker, and J. Hausner. Determining the relation between fire engine travel times and travel distances in New York City. *Operations Research*, 23:614–627, 1975.

[33] J. Krumm, J. Letchner, and E. Horvitz. Map matching with travel time constraints. In *Society of Automotive Engineers (SAE) 2007 World Congress*, 2007.

[34] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang. Map-matching for low-sampling-rate GPS trajectories. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 352–361. ACM, New York, 2009.

[35] F. Marchal, J. Hackney, and K.W. Axhausen. Efficient map matching of large Global Positioning System data sets: Tests on speed-monitoring experiment in Zurich. *Transportation Research Record: Journal of the Transportation Research Board*, 1935:93–100, 2005.

[36] A.J. Mason. Emergency vehicle trip analysis using GPS AVL data: A dynamic program for map matching. In *Proceedings of the 40th Annual Conference of the Operational Research Society of New Zealand*, pages 295–304. Wellington, NZ, 2005.

[37] A.J. Mason. Personal communication, 2012.

[38] M.S. Maxwell, M. Restrepo, S.G. Henderson, and H. Topaloglu. Approximate dynamic programming for ambulance redeployment. *INFORMS Journal on Computing*, 22:266–281, 2010.

[39] D. McFadden. Conditional logit analysis of qualitative choice behavior. In *Frontiers in Econometrics*, pages 105–142. Academic Press, New York, 1973.

[40] L.A. McLay. Emergency medical service systems that improve patient survivability. In *Wiley Encyclopedia of Operations Research and Management Science*. Wiley, New York, 2010.

[41] T. Miwa, D. Kiuchi, T. Yamamoto, and T. Morikawa. Development of map matching algorithm for low frequency probe data. *Transportation Research Part C*, 22:132–145, 2012.

[42] N.J. Nilsson. *Artificial Intelligence: A New Synthesis*. Morgan Kaufmann, San Francisco, 1998.

[43] J.P. Pell, J.M. Sirel, A.K. Marsden, I. Ford, and S.M. Cobbe. Effect of reducing ambulance response times on deaths from out of hospital cardiac arrest: cohort study. *BMJ: British Medical Journal*, 322:1385, 2001.

[44] J.S. Pyo, D.H. Shin, and T.K. Sung. Development of a map matching method using the multiple hypothesis technique. In *Intelligent Transportation Systems Proceedings*, pages 23–27. IEEE, 2001.

[45] M.A. Quddus, W.Y. Ochieng, and R.B. Noland. Current map-matching algorithms for transport applications: State-of-the art and future research directions. *Transportation Research Part C*, 15:312–328, 2007.

[46] M. Rahmani and H.N. Koutsopoulos. Path inference from sparse floating car data for urban networks. *Transportation Research Part C*, 30:41–54, 2013.

[47] H. Rakha and W. Zhang. Estimating traffic stream space mean speed and reliability from dual-and single-loop detectors. *Transportation Research Record: Journal of the Transportation Research Board*, 1925:38–47, 2005.

[48] M. Ramezani and N. Geroliminis. On the estimation of arterial route travel time distribution with Markov chains. *Transportation Research Part B*, 46:1576–1590, 2012.

[49] S. Resnick. *A Probability Path*. Springer, New York, 1999.

[50] R.A. Rigby and D.M. Stasinopoulos. Generalized additive models for location, scale and shape. *Journal of the Royal Statistical Society: Series C*, 54:507–554, 2005.

[51] C.P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer-Verlag, New York, 2004.

[52] G.O. Roberts and J.S. Rosenthal. Optimal scaling for various Metropolis-Hastings algorithms. *Statistical Science*, 16:351–367, 2001.

[53] F. Soriguera and F. Robuste. Estimation of traffic stream space mean speed from time aggregations of double loop detector data. *Transportation Research Part C*, 19:115–129, 2011.

[54] W.E. Stein and R. Dattero. Sampling bias and the inspection paradox. *Mathematics Magazine*, 58:96–99, 1985.

[55] M.A. Tanner and W.H. Wong. The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association*, 82:528–540, 1987.

[56] L. Tierney. Markov chains for exploring posterior distributions. *The Annals of Statistics*, 22:1701–1728, 1994.

[57] N.R. Velaga. Development of a weight-based topological map-matching algorithm and an integrity method for location-based ITS services. *Technical Report, Loughborough University*, 2010.

[58] J.G. Wardrop. Some theoretical aspects of road traffic research. *Proceedings of the Institute of Civil Engineers*, 2:325–378, 1952.

[59] H. Wei, Y. Wang, G. Forman, and Y. Zhu. Map Matching by Fréchet Distance and Global Weight Optimization. *Working paper*, 2013.

[60] B.S. Westgate, D.B. Woodard, D.S. Matteson, and S.G. Henderson. A Monte Carlo Method for Map-Matching, with GPS Bias Estimation. *Working paper*, 2013.

[61] B.S. Westgate, D.B. Woodard, D.S. Matteson, and S.G. Henderson. Large-network travel time distribution estimation, with application to ambulance fleet management. *Under review*, 2013.

[62] B.S. Westgate, D.B. Woodard, D.S. Matteson, and S.G. Henderson. Travel time estimation for ambulances using Bayesian data augmentation. *Annals of Applied Statistics*, to appear, 2013.

[63] C.E. White, D. Bernstein, and A.L. Kornhauser. Some map matching algorithms for personal navigation assistants. *Transportation Research Part C*, 8:91–108, 2000.

[64] M.G. Wing, A. Eklund, and L.D. Kellogg. Consumer-grade global positioning system (GPS) accuracy and reliability. *Journal of Forestry*, 103:169–173, 2005.

[65] T.H. Witte and A.M. Wilson. Accuracy of non-differential GPS for the determination of speed over ground. *Journal of Biomechanics*, 37:1891–1898, 2004.

[66] H. Xu, H. Liu, C. Tan, and Y. Bao. Development and application of an enhanced Kalman filter and Global Positioning System error-correction approach for improved map-matching. *Journal of Intelligent Transportation Systems*, 14:27–36, 2010.

[67] H. Yin and O. Wolfson. A weight-based map matching method in moving objects databases. In *Proceedings of the 16th International Conference on Scientific and Statistical Database Management*, pages 437–438. IEEE, 2004.